



Integrated Payment Gateway Manual Document

Merchant Integration Support

Version: 2.0

Prepared by: Merchant Integration Team

Email: merchant-integration@etisalat.ae

Updated: June 2015

Table of Contents

IMPORTANT CONTACTS	6
Comtrust Support (24/7).....	6
PKI Support	6
Operations	6
Merchant Integration Support (7 AM – 3 PM).....	6
1. INTRODUCTION.....	7
1.1. Related Documents.....	8
1.2. Glossary of Terms.....	8
2. TRANSACTION TYPES	13
2.1. Point of Sale (POS)	14
2.2. Web Based Payments	14
2.3. Mail Order / Telephone Order (MOTO)	15
2.4. Cardholder Activated Terminal (CAT)	15
2.5. Recurring Payments	16
2.6. BizDirect (Direct Debit Payments)	16
2.7. Any Device Capable to Integrate SPI.....	16
3. PAYMENT MODELS	17
3.1. Authorization Model	18
3.1.1 Card Not Present (Card Not Present Scenario)	18
Introduction	18
Implementation	19
3.1.1.1. Auto Capture.....	19
3.1.1.2 Manual Capture	19
SPI Calls	21
3.1.2 Card Present (Card Present Scenario).....	21
Introduction	21
Implementation	21
3.1.2.1. Auto Capture.....	21

3.1.2.2. Manual Capture	23
SPI Calls	24
3.2. Redirection Model	24
3.2.1. Auto Capture (Card Not Present Scenario)	25
Introduction	25
Implementation	26
SPI Calls	26
3.2.2. Manual Capture (Card Not Present Scenario)	27
Introduction	27
Implementation	27
SPI Calls	29
3.3. 3D-Secure Model	29
Implementation	30
3.4. Recurring Payments Model.....	30
3.4.1 Step-1- Registering Credit Card for recurring payments	31
3.4.2 Step-2- Recurring Payment through Authorization Call or Registration Call.....	31
3.4.2.1 Option-1- Recurring Payment through Registration Call for a registered Credit Card	31
3.4.2.2 Option-2- Recurring Payment through Authorization Call for a registered Credit Card	32
3.5. Comparison between Manual Capture and Auto Capture	33
4. IPG INTEGRATION GUIDE	35
4.1. Payment Specific Decision	36
4.1.1. Web-based payments decisions	36
4.1.1.1. Integrated Payment Portal.....	36
4.1.1.2. Merchant payment entry page using authorization API.....	36
4.1.1.3. SPILess Payment.....	37
4.1.2. MOTO	38
4.1.2.1. Standard authorization call	38
4.1.2.2. Customer Activated Terminals (CAT)	38
4.1.2.3. Recurring transactions	38

4.2.	Merchant Specific Decision.....	40
4.2.1.	Selecting the right capture mode and handling finalization and delivery errors	40
4.2.2	SPI Platform Selection.....	41
4.2.3.	Enabling Verification Code.....	41
4.2.4.	Enabling 3D Secure Authentication	41
5.	SPI USER GUIDE.....	42
	Download Link:	42
5.1.	SPI Installation Guide	43
5.1.1.	Requirements.....	43
5.1.1.1.	Firewall/Router Configuration	43
5.1.1.2.	Connectivity Testing.....	43
5.1.1.3.	Digital Certificate.....	43
5.1.2.	SPI.net	44
5.1.2.1.	System Requirements	44
5.1.2.2.	Installation	44
5.1.2.1.	SSL Requirements.....	44
5.1.2.2.	Configuration	44
5.1.2.2.1.	Loading from configurations file.....	45
5.1.2.2.2.	Manual configurations.....	45
5.1.3.	SPI.j.....	46
5.1.3.1.	System Requirements	46
5.1.3.2.	Installation	47
5.1.3.3.	SSL Requirements.....	47
5.1.3.4.	Configuration	47
5.1.4.	SPIs (Windows Service)	48
5.1.4.1.	System Requirements	48
5.1.4.2.	Installation	48
5.1.4.3.	SSL Requirements.....	48
5.1.4.4.	Configuration	48

5.1.5.	SPI Less	48
5.2.	SPI Calls	50
5.2.1.	Registration	50
5.2.2.	Finalization	51
5.2.3.	Authorization	52
5.2.4.	Capture	54
5.2.5.	Reversal	55
5.3.	SPI Transaction Properties	56
5.3.1.	Connection Properties	56
5.3.2.	SPI Specific Connection Properties	56
5.3.3.	Point of Sale Properties	57
5.3.4.	Transaction Properties	57
5.3.5.	Buyer Properties	60
5.3.6.	Response Properties	60
5.3.7.	Customer Properties	61
5.3.8.	Property types	62
5.3.8.1.	Early bind properties	62
5.3.8.2.	Late Bind Properties	63
6.	Configurations and Test Data for Testing in IPG Staging	64
6.1.	Test Configurations	65
6.2.	Test Data	65
7.	FAQs AND KNOWLEDGEBASE	66
8.	MERCHANT PORTAL	67
9.	MERCHANT INTEGRATION TEST CASES	68
	Use Case 1: Merchant Configuration 'Registration' confirmation on IPG Staging Server	69
	Use Case 2: Common Payment Page (CPP) appears with correct settings and values	70
	Use Case 3: Merchant Configuration 'Finalization' on IPG Staging Server	70
	Use Case 4: Capture/Reversal on IPG Staging Server	71
	Use Case 5: Verify closing of a transaction on IPG Staging Server	72

IMPORTANT CONTACTS

Comtrust Support (24/7)

support@comtrust.ae | 800-2900

Comtrust Support is the first level support for Etisalat Payment Gateway related issues. This team is available 24/7 for any issues related to IPG production environment and should be contacted in the first place in case any live customer gets a payment related issue. Once contacted this team is responsible to rectify the issue and forward it to related team for resolution. This support group has to be contacted for merchant provisioning, updating merchant settings or any issues related to IPG Production server.

PKI Support

pkihelp@eim.ae – pkira@etisalat.ae

PKI Support group deals with the issues related to Digital Certificate issuance/installation.

Operations

Ops-team@eim.ae

Operations Team is the second level support for IPG and is responsible for maintaining and monitoring IPG Production server.

Merchant Integration Support (7 AM – 3 PM)

Merchant-integration@etisalat.ae

Merchant Integration Support group is responsible for the configuration and integration of IPG merchants into the IPG. This group also deals with any issues raised for IPG Staging server. This group is the third level of support for IPG and handles the cases for Production server only when contacted by first or second level support. Availability is strictly between office hours (7 AM – 3 PM, UTC+04:00).

1. INTRODUCTION

Integrated Payment Gateway (IPG) offers customers a comprehensive payment collection solution. It is a proprietary payment platform that enables real-time authorization of payment transactions from Visa, MasterCard, Diners Club, JCB and American Express. This easy and quick-start package provides a reliable, affordable and secure payment mechanism for ecommerce retailers. In addition, the platform is able to accept and process debit cards, BizDirect (Direct Debit), eDirham transactions as well as other customized payment instruments.

1.1. Related Documents

<https://demo-ipg.comtrust.ae/merchant/downloads/IPGKnowledgeBase.pdf>

1.2. Glossary of Terms

The table below consists of a compiled list of terms that will assist the reader in understanding the jargon used extensively in the field of electronic commerce.

Terms	Description
3D Secure Authentication	New system aimed at reducing online credit card fraud and chargeback. It enables additional authentication on the cardholder's identity by asking for additional PIN during an online shopping transaction. VISA name it "Verified by VISA" while MasterCard name it "Master SecureCode".
Acquirer	The bank which recruits shops and other service providers to accept payment cards. Acquirers process a merchant's transactions and pass them into the clearing system to allow financial settlement.
Authentication	The process of verifying the identity and legitimacy of a person, object or system.
Authorization	The process whereby a merchant (or a cardholder through an ATM) requests permission from the Card Issuer for the card to be used for a particular transaction.
Captured	The status of a transaction that has gone through Authorization or Post Authorization and is waiting for settlement.
Card Issuer	The bank or company which issues a payment card to the customer, and which has financial responsibility for a card originated transaction.
Card Verification Code	The last three or four digits of a number printed on or just below the signature panel or on the front of payment cards.
Cardholder	A person to whom a payment card has been issued.
Cardholder Activated Terminal (CAT)	A terminal activated by the cardholder and not supervised by a member of staff on behalf of the merchant. May also be referred to as an Unattended Payment Terminal (UPT).

Card Not Present Scenario	A transaction where the merchant does not have physical access to the card (e.g. through telephone, mail order or Internet transactions). All transactions where a credit card is not physically swiped through a terminal, including internet transactions, phone transactions, or credit-card numbers keyed into a terminal/virtual terminal, fall into this category.
Card Present Scenario	A transaction where the card is presented physically to the merchant. Examples are PoS transactions, online transactions where Secure Code is presented etc.
Common Payment Page(CPP)	In compliance with 3D Secure security guidelines, IPG provides a common payment portal, which can be used by the merchant, for redirecting the card-holder once he is ready to provide the credit card details. The information is then taken by IPG, through the Merchant Plug-in and passed to the Issuer for Authentication. A Merchant logo can be used on this payment portal to identify the merchant. The Merchant can modify design (fonts, color schema, layout) of the Payment Portal to make it appear to be a part of the merchant's website.
Credit Card	A payment card that enables the holder to make purchases and to draw cash up to a pre-arranged limit.
Credit Card Processing	Once the payment gateway accepts the transaction, this service records the transaction, removes funds from the credit cardholder's account and deposits these into the merchant account.
CVC2 & CVV2	Card Validation Code 2 & Card Verification Value 2. VISA and MasterCard implemented a security code feature known as "CVV2" and "CVC2". The security code is a 3 digit code imprinted on the physical credit card, but not embedded or encrypted in the magnetic stripe. The code is a 3-digit number located on the signature strip on the back of Visa and MasterCard cards, after the card number. This three-digit code helps validate that the cardholder has the card in his/her possession, and the account is legitimate.

Debit Card	A payment card linked to a bank account, used to pay for goods and services by debiting the holder's account, usually also combined with other facilities such as ATM and cheque guarantee functions.
Digital Certificate	A digital certificate is an electronic certificate that establishes credentials when performing transactions on the Web. The certificate is issued by a Certification Authority in this case IPG.
Electronic Funds Transfer (EFT)	EFT is a technology (one of the electronic commerce technologies) that allows the transfer of funds from the bank account of one person or organization to that of another. EFT is also used to refer to the action of using this technology. It is an important addition in the organization that implements EDI in their organization.
Encryption	A method of ensuring data secrecy. The message is coded using a key available only to the sender and the receiver. The coded message is sent to the receiver and then decoded upon receipt.
Firewall	A computer system that sits between the Internet and a company's LAN. It is a means of automatically limiting what a company's computer system will pass along to outside computer systems. It acts as an active gateway to keep non-company entities from accessing company confidential data.
Gateway	Most generally, a computer that forwards and routes data between two or more networks of any size.
MasterCard SecureCode	MasterCard SecureCode is a new service to enhance the user's existing MasterCard account. A PIN code will be linked to the credit card for added protection against unauthorised use of the card when dealing with online merchants. Similar technology by VISA is known as VERIFIED by VISA.
Merchant	The organization (the Departments of Abu Dhabi) accepting credit card payments for the services they provide. The term merchant in this On-boarding guide will be used to describe the role and activities needed to be taken up by the prospective Department of Abu Dhabi interested in implementing IPG Payment Solutions.

Merchant Bank Account	Allows an organization to accept credit card transactions from customers. Merchant bank accounts are commercial bank accounts set up between an organization and a financial institution. Funds from customers are deposited into the merchant account.
Payment Gateway	A computer system that acts as a mediator between a merchant account and online storefront. Payment gateway is used in authentication of credit card information and real-time charging from a credit card.
Point-Of-Sale (PoS)	(or Point of Service) Location where a customer makes a purchase.
PoS Terminal	An electronic device used to accept and process card payments at point-of-sale.
Post Authorization	A transaction that puts a Pre Authorization transaction into a Captured state for settlement. In the case of partial shipments, the Post Authorization amount may be less than the Pre Authorization amount.
Pre Authorization	A transaction type in which a cardholder's account is verified to be in good standing, that is, the card is valid and has not reached its limit. If the verifications are approved, the total amount of the order is reserved against the cardholder's account balance. Pre Authorization is used if goods are to be physically shipped or in other cases for which the merchant must first verify whether the order can be fulfilled. An approved Pre Authorization is followed by a Post Authorization, which prepares it for settlement.
Request for information (RFI)	Where an issuer requests an acquirer to supply details of a cardholder's specific transaction undertaken at a point-of-sale device.
Secure Sockets Layer (SSL)	It's a protocol designed for secure transfer of Information over the Internet. Information sent through an SSL-secured form is encrypted so that the information is not tempered with while the transfer is taking place.
Verified by Visa (VbV)	Verified by Visa is a system used by Visa as an added layer of security for online credit card transactions. It relies on a password to validate the transaction. This acts in the same way as using a PIN or signature when purchases are made over the counter. This will ensure that it is

	in fact the actual cardholder making the purchase. The similar system is used by Master Card under the name SecureCode. It is an easy-to-use password-protected online payment verification system that ensures that the cardholder is who they say they are.
Work Order	Work Order in other words is the service order containing all the information related the merchant configuration, acquirer bank and other configuration and contractual information.

2. TRANSACTION TYPES

The Internet has become a vital part of people's lives, providing more and more consumers with the option to shop, pay bills, bank and invest online through the use of electronic payments (ePayments). Electronic Commerce or eCommerce in short, consists of any transaction where the buying and selling of products and services is conducted over electronic channels. In most cases these transactions involve electronic payment which could take any form of Electronic Fund Transfer (EFT), including credit card payment, direct debit and even standing instructions.

This section describes types of electronic payment transactions that are critical in deciding the ePayment Solution best suited for a Merchant.

- 1. Point of Sale (PoS)**
- 2. Web Based Payments**
- 3. Mail Order / Telephone Order (MOTO)**
- 4. Card Activated Terminal**
- 5. BizDirect (Direct Debit Payments)**
- 6. Any Device Capable to Integrate SPI**

2.1. Point of Sale (POS)

Most common form of electronic payment transaction is POS mode. In this mode, the payer has to be physically present at the time of transaction. In addition, the payer has to physically hold payment token (this is usually in form of a card, although contactless payment tokens can be used). In most cases, POS transaction requires either a signature or an electronic pin hence POS terminals are manned. While payment happens through electronic means, POS are seldom used in eCommerce applications as manned terminals remove most of benefits of electronic fulfillment. POS terminals are usually connected directly to the acquirer bank, but in big organizations they could be linked through a payment gateway which in turn provides centralised rules, reconciliation and reporting.

2.2. Web Based Payments

The most popular electronic transaction type in the eCommerce landscape is web-based payment. It provides the widest reach, with the most user-friendly and attractive user interfaces, combined with the ability for the payer to securely hand-over important information directly to the trusted party. Conversely, majority of web-based transactions depend solely on information printed on the payment token (as only information known to payer could be used). This provides an opportunity for unauthorised usage of the payment token information for fraudulent activities, in the situation when this information is acquired by someone purporting to be the payer. Furthermore, the anonymous nature of the media (Internet) makes it very difficult to have independent information on the origin of the transaction. While IP addresses may play a role, there are multiple situations where they do not bring additional value.

Over time, several approaches were developed to make web-based transactions more secure with two main streams. The first one involved the securing of web-based transactions without direct input / help from issuer of the payment token - those systems called fraud detection or fraud monitoring systems had the capability to either verify extra data not printed on the card (i.e. card issuer name, country of issuance) or crosscheck the provided information with other data known (i.e. IP address, shipping address). These systems can analyze transaction behavior (e.g. number of transactions with the same card) and eliminate issuers, countries, systems (i.e. open proxies) where most fraud is known to be originated or where legislation supporting merchants in seeking legal recourse against fraudsters is absent. The most advanced systems offered score based systems, where each transaction could be scored against multiple criteria or even elements of artificial intelligence.

The other stream is based on information directly verified by issuer of payment token - this could be card holder name, billing address or any other information known to issuer. Unfortunately, in the long run, the information collected by merchant proved to affect the payer's privacy, in addition to the failure of issuers to be able to verify such information across the world. The only solution to keep such data private was for the payer to enter it directly onto the issuer website (which should be the only party besides the payer who will have such information). In this case, security of web-based payment transactions should be considered similar to the one used while accessing eBanking websites or using ATMs. The family of protocols allowing merchant to benefit from such security is commonly called 3Dsecure, however different payment schemas use their own names like VbV (Verified by Visa), SecureCode and J-Secure. The addition of 3Dsecure protocols significantly improved the security of web-based transactions and in the case where the merchants fulfill all the requirements of such a programme; they are no longer liable for fraudulent transactions (although there are some exceptions to this).

2.3. Mail Order / Telephone Order (MOTO)

Recurring payment transactions are not constantly initiated by payer (and are not in the presence of payer like in case of POS mode). Such payments takes place for orders received by phone, email / mail, as well as those where the payer provided his payment details and allowed merchant to charge him periodically (e.g. service payments, installments, etc.). While most of original requirement for MOTO transactions can be now supported by the previously explained transaction types, call centers and recurring payments remain major users of this mode.

MOTO transactions can be further divided to those where payment details are handed to the merchants and those where the merchant processes the transaction without knowing the card number. The latter is possible for a recurring transaction and only if certain other conditions are met. Such recurring transactions can be used for both regular transactions with fixed amounts (classical installments) or for ad hoc transaction with variable amounts, in which case the merchant benefits from payment gateway card storage.

2.4. Cardholder Activated Terminal (CAT)

CAT transaction mode is very similar to the POS mode explained in earlier. The key difference is that the CAT transaction is initiated by the cardholder and does not require the presence of a merchant representative. Thus it fits well into a kiosk mode where the standalone machine is placed in public place and allows the cardholder to make payments without the usage of Internet and / or mobile. CAT transactions are applicable to payment cards only and they require additional device (magnetic stripe reader) to be embedded into the KIOSK itself as the transaction

is made by swiping card rather than entering its details. CAT mode is popular for simple services like billing or penalty payments as well in unmanned physical environments - i.e. self-service petrol stations.

2.5. Recurring Payments

Recurring Payments is a solution where a Merchant wants to save Credit Card information (sensitive data) and payer gives him permission to make future transactions on his behalf or may be on just one click by payer (payer does not need to provide same Credit Card information again and again) for his future transactions.

As per PCI standards, only PCI compliant companies are allowed to store any Credit Card sensitive data like card number. Every merchant who wants to implement Recurring Payments kind of functionality cannot afford to be PCI compliance.

So IPG is providing “Recurring Payments” feature where merchants will redirect the payers to IPG where they’ll provide their Credit Card details, IPG will authenticate provided data and return a reference for saved Card details for future Recurring Payment call from same Merchant for that specific Credit Card.

2.6. BizDirect (Direct Debit Payments)

BizDirect is a payment mechanism that would offer individuals and corporate users the ability to pay for ecommerce services directly from their bank account. This payment can be performed from within the secure internet banking environment available from their bank.

2.7. Any Device Capable to Integrate SPI

Generally any device that can integrate SPI for example devices running Java VM or operating on Microsoft Windows may easily integrate SPI and hence can make payments.

3. PAYMENT MODELS

IPG provides two models for making electronic payments, depending on presence of payment instrument i.e. Visa Card etc.

1. Authorization Model

Authorization Model refers to the traditional way for making online payments, where Merchant (seller) collects all the credit card related information from his Customer (buyer) and sends it to the Payment Gateway for processing. This model is also adapted for payments where redirection is not an option like POS terminals, CAT transactions, MOTO transactions and other SPI enabled devices.

2. Redirection Model

Redirection Model is a more secure way considered for online payments where Merchant (seller) redirects his Customer (buyer) to Payment Gateway provided page (exposed on Payment Gateway network), buyer provides his credit card related information to Payment Gateway directly and is redirected to seller website after authentication and other checks by Payment Gateway. This model is secure as credit card data is not handed over to any untrusted party; according to PCI standards, any non PCI compliant party cannot save credit card information.

3. 3D Secure

In compliance with 3D Secure security guidelines, IPG provides a common payment portal, which can be used by the merchant, for redirecting the card-holder once he is ready to provide the credit card details. The information is then taken by IPG, through the Merchant Plug-in and passed to the Issuer for Authentication. A Merchant logo can be used on this payment portal to identify the merchant.

4. Recurrence Model

In compliance with PCI standards, any non PCI compliant party cannot save Credit Card information. So in order to assist Merchants (non PCI compliant) IPG provides this feature to save payer's Credit Card information in IPG for making recurring transactions in future.

3.1. Authorization Model

Authorization Model refers to the traditional way for making online payments, where Merchant (seller) collects all the credit card related information from his Customer (buyer) and sends it to the Payment Gateway for processing. This model is also adapted for payments where redirection is not an option like POS terminals, CAT transactions, MOTO transactions and other SPI enabled devices.

Figure 1 shows the business workflow of a transaction following Authorization Model.

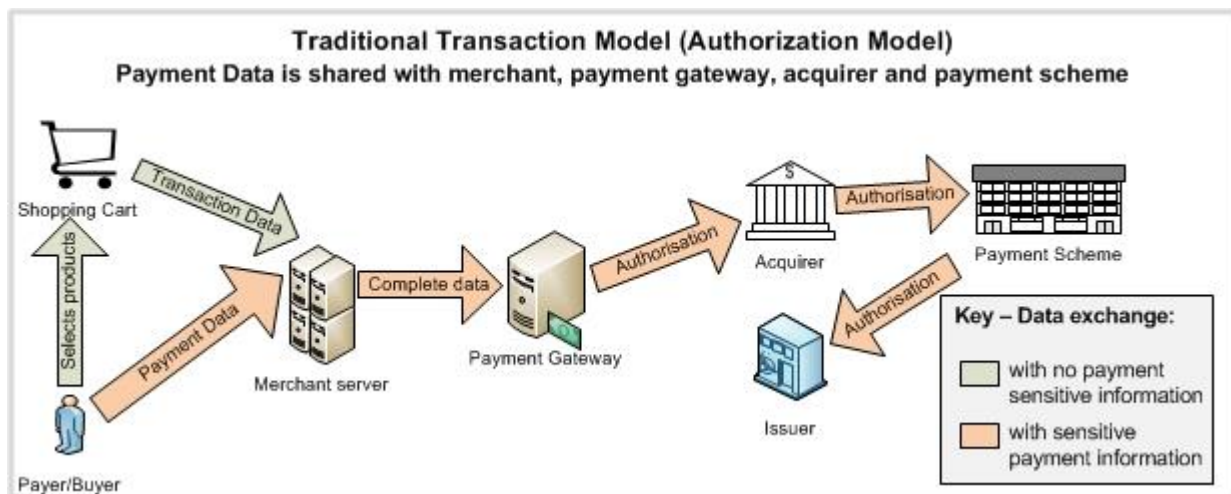


Figure 1

Authorization Model in IPG can be adapted in two ways:

3.1.1 Card Not Present (Card Not Present Scenario)

Introduction

Card Not Present payments take place for orders received by phone, email/mail, as well as those where the payer provided his payment details and allowed merchant to charge him periodically (e.g. service payments, installments, etc.) or in the scenario where Merchant wants to receive credit card information from the customer on his web site.

These transactions can be further divided to those where payment details are handed to the merchants and those where the merchant processes the transaction without knowing the card number. The latter is possible for a recurring transaction and only if certain other conditions are met. Such recurring transactions can be used for both regular transactions with fixed amounts (classical installments) or for ad hoc transaction with variable amounts, in which case the merchant benefits from payment gateway card storage.

Implementation

Merchants can integrate to IPG for a Card Not Present transaction using SPI (can be downloaded from <https://demo-ipg.comtrust.ae/merchant/#/downloads>).

Following are the two implementation models for Card Not Present scenario:

3.1.1.1. Auto Capture

Figure 2 explains the technical work flow of a Card Not Present transaction with Auto Capture.

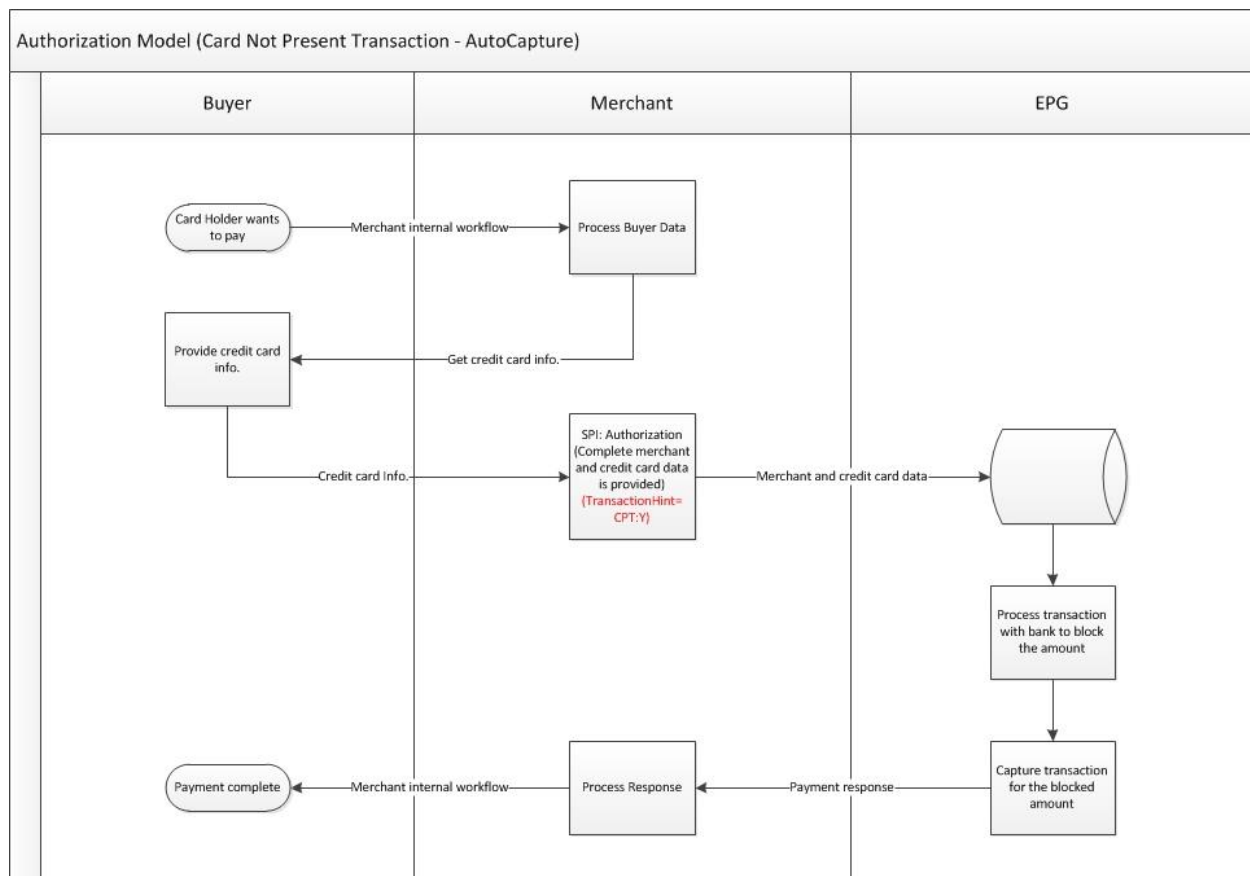


Figure 2

3.1.1.2 Manual Capture

Figure 3 explains the technical work flow of a Card Not Present transaction with Manual Capture.

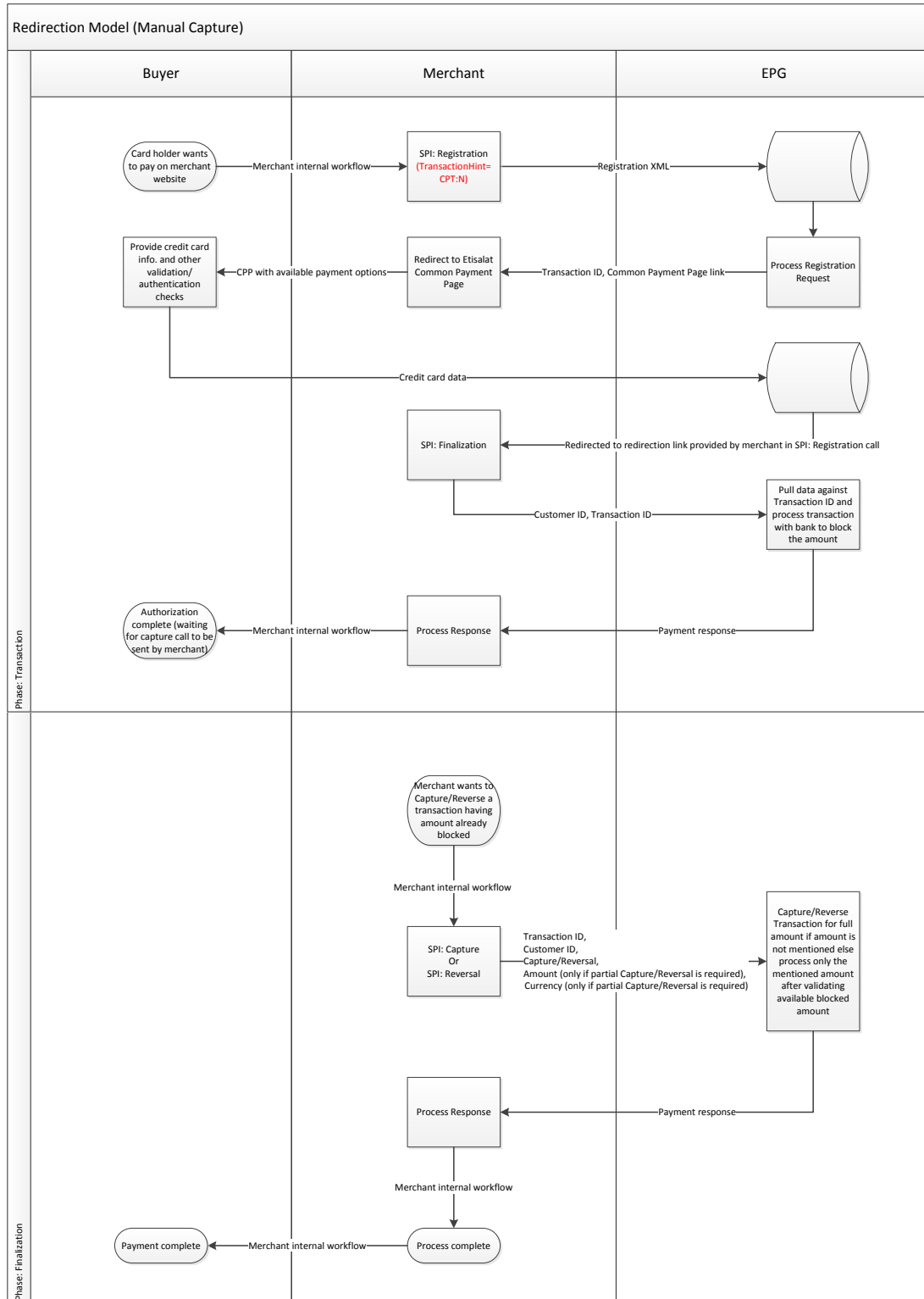


Figure 3

For SPI documentation, please refer to Section 5. *SPI USER GUIDE* for calls included in Card Not Present transactions. Please read the manual for SPI Authorization, Capture and Reversal calls.

Authorization call takes all Merchant and Credit Card related data, processes the payment with the payment parties and returns the response. Complete processing is done in single call.

SPI Calls

- a. Authorization
- b. Capture / Reversal (required only in case Manual Capture has been configured)

Please refer to section 5.2 *SPI Calls* for any help and documentation related to above mentioned SPI calls.

Note: Please strictly follow the calling sequence.

3.1.2 Card Present (Card Present Scenario)

Introduction

Card Present payments are valid where card is present physically and is swiped into the machine. Thus it fits well into POS machine transaction or a kiosk transaction where the standalone machine is placed in public place and allows the cardholder to make payments without the usage of Internet and/or mobile. Card Present transactions are applicable to payment cards only and they require additional device with magnetic stripe reader to be embedded into the KIOSK itself or a POS machine as the transaction is made by swiping card rather than entering its details. CAT mode is popular for simple services like billing or penalty payments as well in unmanned physical environments - i.e. self-service petrol stations.

Implementation

Merchants can integrate to IPG for a Card Present transaction using SPI (can be downloaded from <https://demo-ipg.comtrust.ae/merchant/#/downloads>).

Following are the two implementation models for Card Not Present scenario:

3.1.2.1. Auto Capture

Figure 4 explains the technical work flow of a Card Present transaction with Auto Capture.

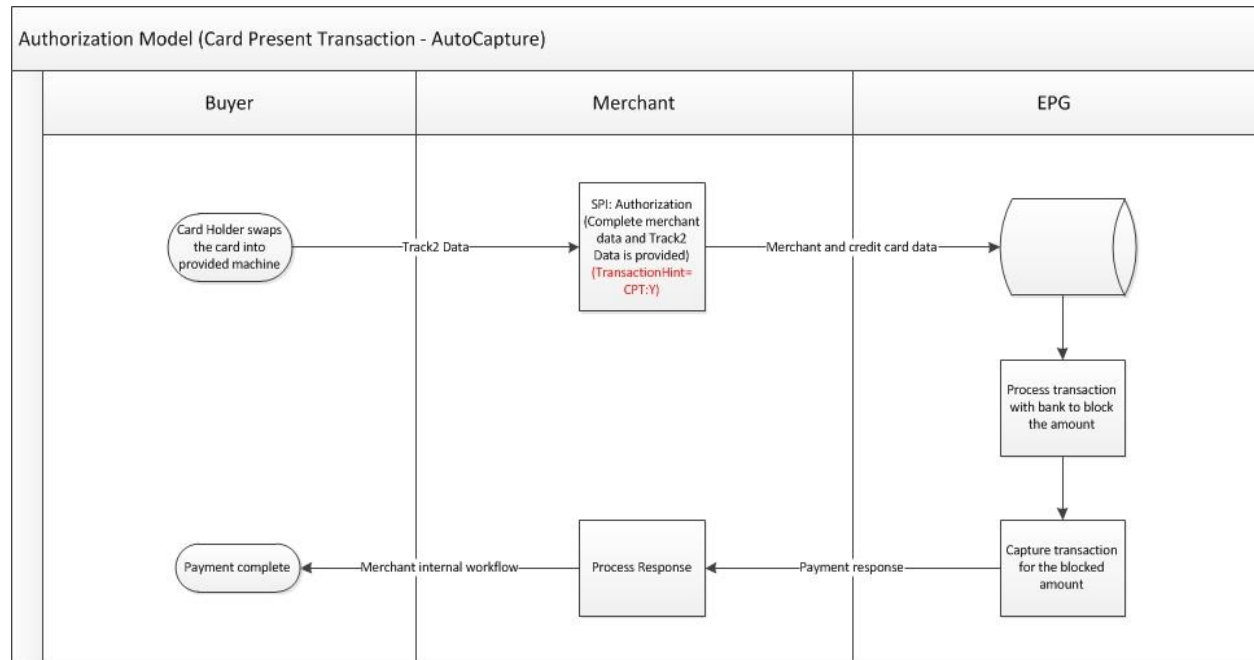


Figure 4

3.1.2.2. Manual Capture

Figure 5 explains the technical work flow of a Card Present transaction with Manual Capture.

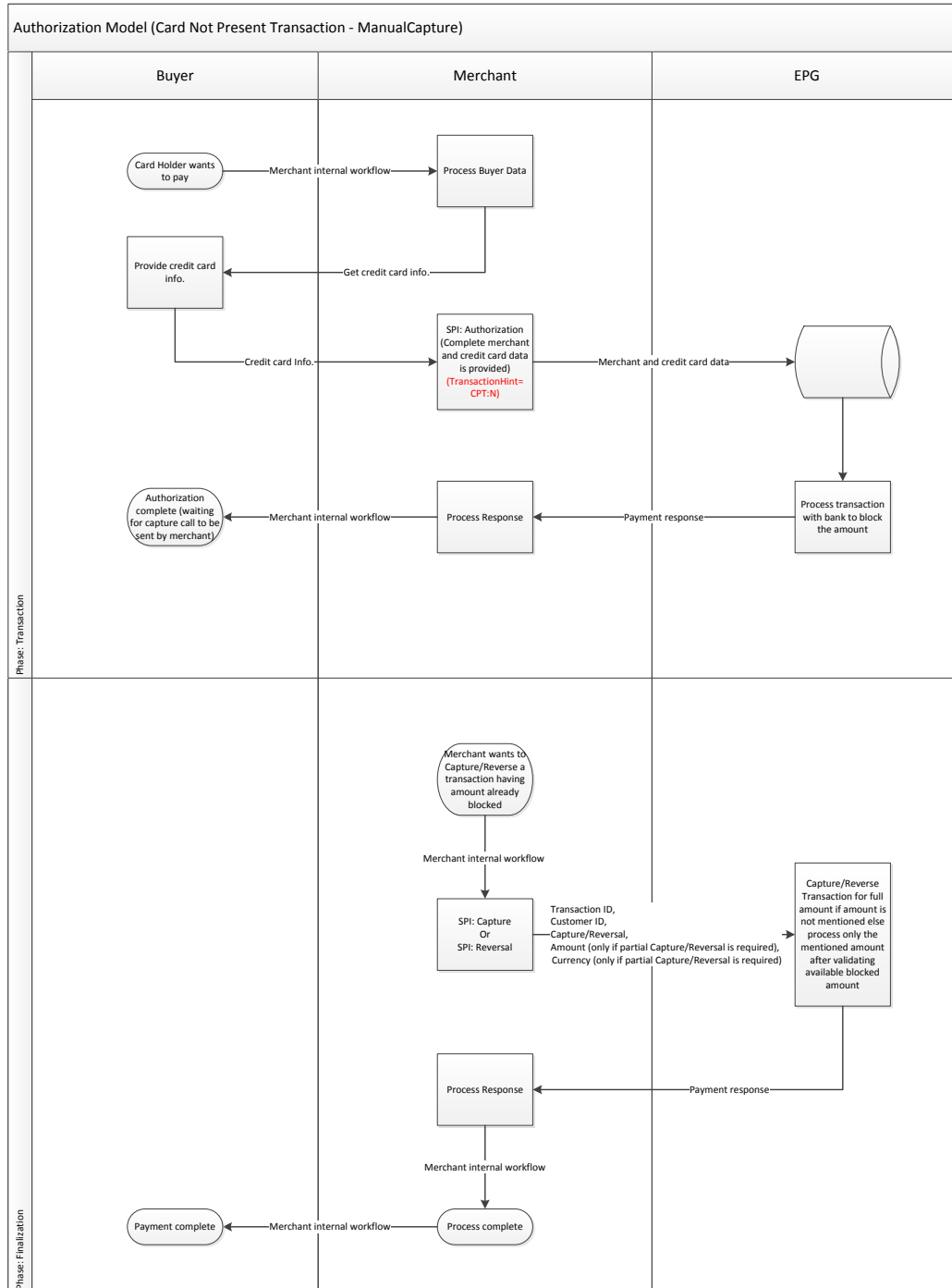


Figure 5

For SPI documentation, please refer to Section 5. *SPI USER GUIDE*. Please read the manual for SPI Authorization, Capture and Reversal calls.

For Card Present transaction, we need to implement Authorization call with a little change which is as follows:

- Ignore all Credit Card related fields (CardNumber, ExpiryYear, ExpiryMonth and SecureCode)
- Provide Credit Card Track 2 Data in property **Track2Data**

Authorization call takes all Merchant and Credit Card related data, processes the payment with the payment parties and returns the response. Complete processing is done in single call.

SPI Calls

- a. Authorization
- b. Capture / Reversal (required only in case Manual Capture has been configured)

Please refer to section 5.2 *SPI Calls* for any help and documentation related to above mentioned SPI calls.

Note: Please strictly follow the calling sequence.

3.2. Redirection Model

Redirection Model is a more secure way considered for online payments where Merchant (seller) redirects his Customer (buyer) to Payment Gateway provided page (exposed on Payment Gateway network), buyer provides his credit card related information to Payment Gateway directly and is redirected to seller website after authentication and other checks by Payment Gateway. This model is secure as Payment Instrument (for example Credit Card) related data is not handed over to any untrusted party; according to PCI standards, any non PCI compliant party cannot save credit card information.

Following steps take place in context of a buyer when Redirection Model has been implemented:

1. Buyer verifies on Merchant's web site that he wants to pay for the purchases
2. Merchant's web site redirects the buyer to secure and trusted Payment Page provided by IPG
3. Buyer provides Payment Instrument related information to IPG Common Payment Page (CPP)

4. On authentication and verification CPP redirects the buyer back to Merchant's web site and Merchant displays the status of transaction to the customer

Hence enforcing that no Payment Instrument related information has been provided to and/or kept by the Merchant (non PCI compliant party).

Figure 6 shows the business workflow of a transaction following Redirection Model.

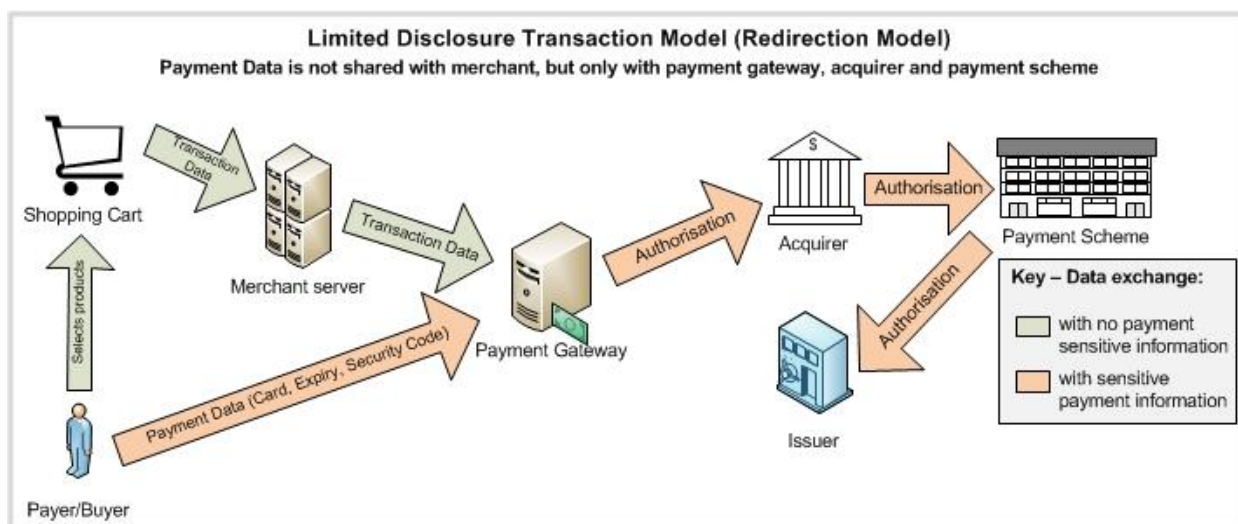


Figure 6

Redirection Model in IPG can be adapted in two ways (implementation of suitable method is important in Merchant's business context only – explained below):

3.2.1. Auto Capture (Card Not Present Scenario)

Introduction

Auto Capture allows the Merchant to process the transaction without any delays between blocking of the amount from payer's Payment Instrument and capturing it to finally process the transaction with bank. In this scenario Merchant is telling IPG that he has already verified the transaction and IPG should process it immediately. It automatically closes a transaction after successful payment.

Auto Capture is recommended if your business flow does not need any physical/logical authentication/validation, for example, a retail store.

Implementation

Merchants can integrate to IPG for Auto Captured transaction using SPI (can be downloaded from <https://demo-ipg.comtrust.ae/merchant/#/downloads>).

Figure 7 explains the technical work flow of an Auto Capture transaction.

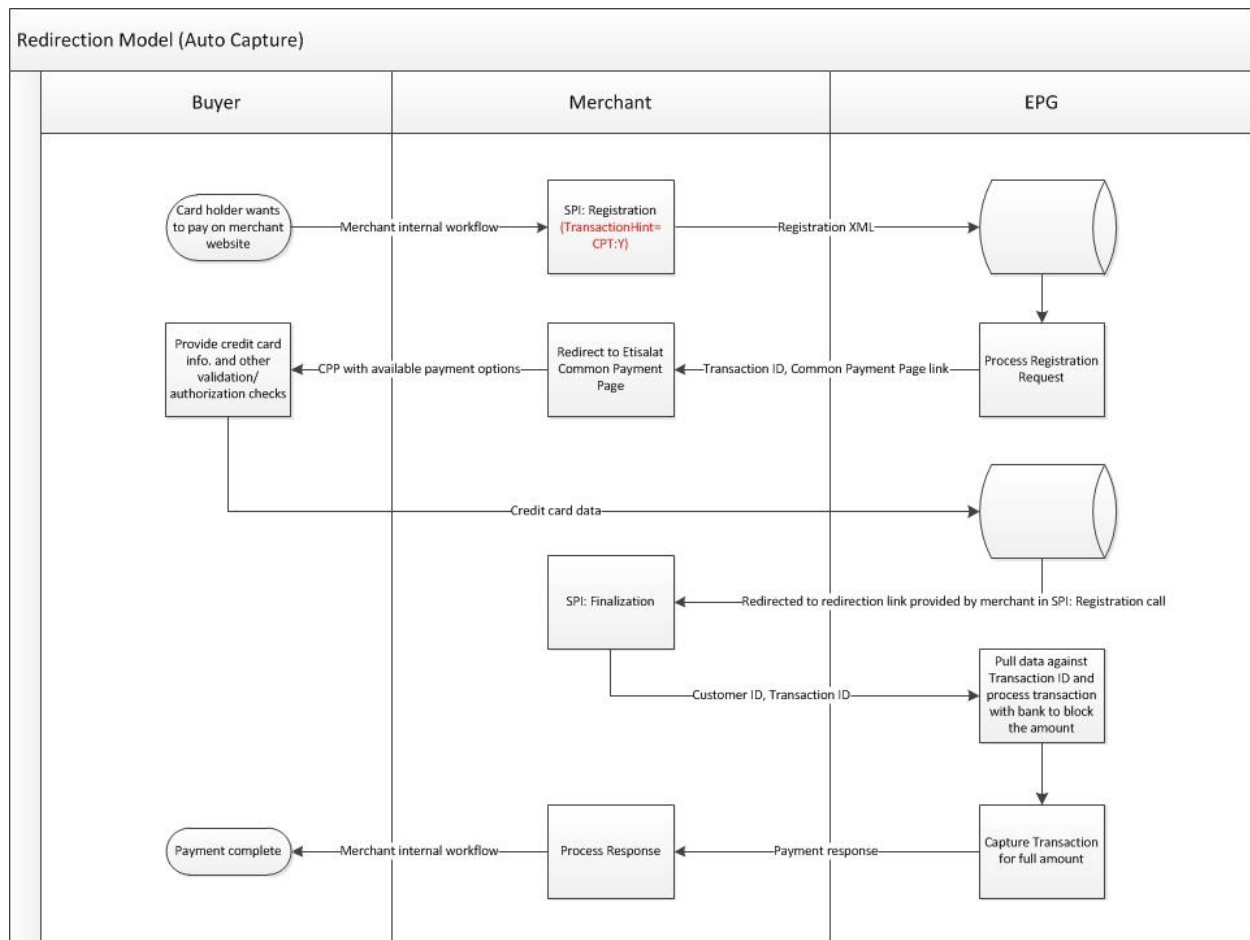


Figure 7

This model includes two calls to IPG using SPI. For SPI documentation, please refer to Section 5. *SPI USER GUIDE*. Please read the manual for SPI Registration and Finalization calls. In SPI downloads package a sample can be found for Auto Capture transactions.

SPI Calls

- Registration
- Finalization



Please refer to section 5.2 *SPI Calls* for any help and documentation related to above mentioned SPI calls.

Note: Please strictly follow the calling sequence.

3.2.2. Manual Capture (Card Not Present Scenario)

Introduction

Manual Capture allows the Merchant to only block the amount mentioned in Registration. To process and close the transaction Merchant needs to initiate Capture or Reversal call separately after successful Finalization call. This process gives Merchant chance to validate the order before processing the payment.

Manual Capture is used if business requires any physical/logical authentication/validation like in case you have to validate a physical delivery address or amount has to be captured subject to goods received.

Implementation

Merchants can integrate to IPG for Manual Captured transaction using SPI (can be downloaded from <https://demo-ipg.comtrust.ae/merchant/#/downloads>).

Figure 8 explains the technical work flow of a Manual Capture transaction.

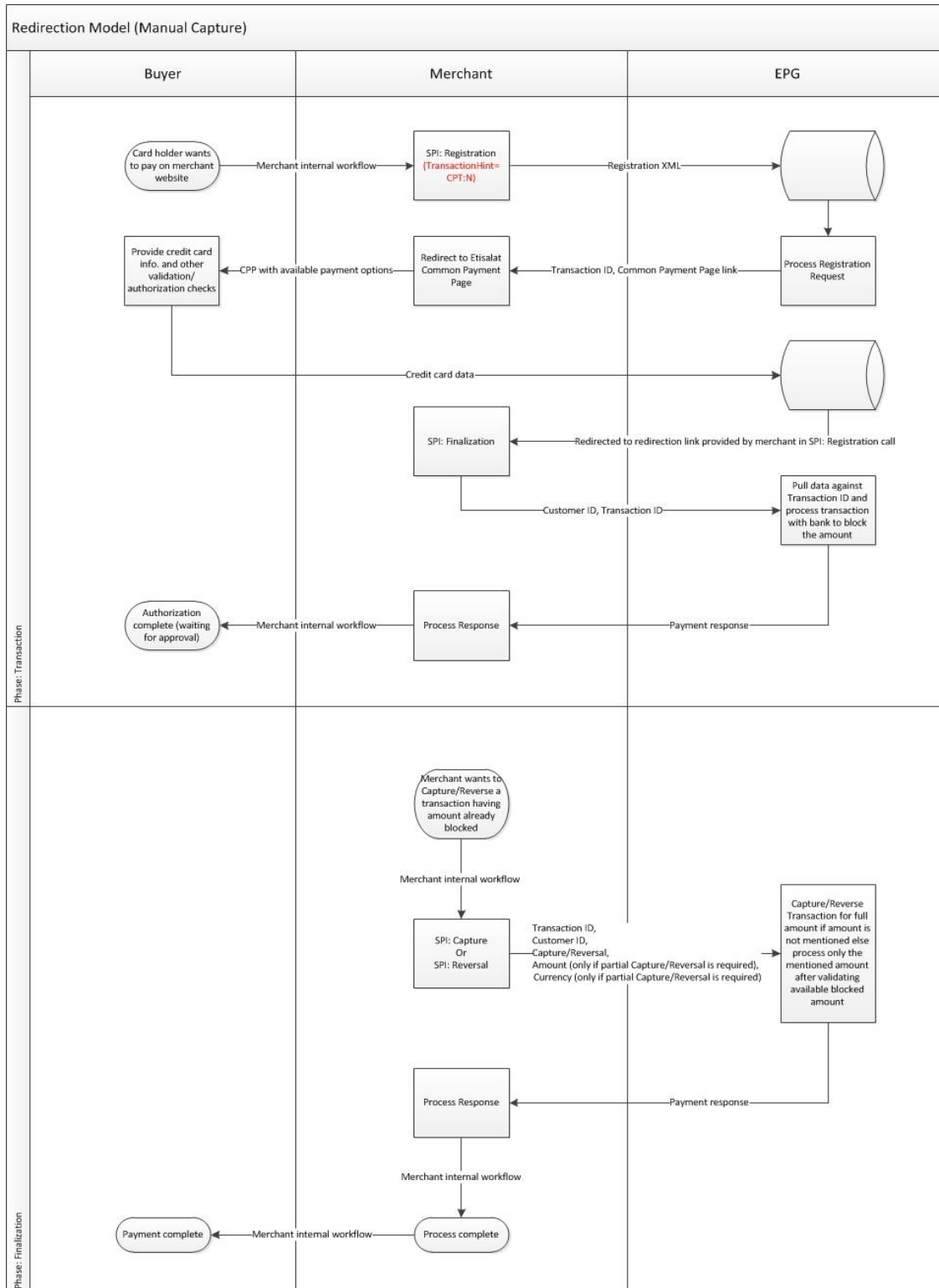


Figure 8

This model includes three calls (minimum) to IPG using SPI. For SPI documentation, please refer to Section 5. SPI USER GUIDE. Please read the manual for SPI Registration, Finalization, Capture (see partial capture as well), Reversal (see partial reversal as well) calls. In SPI downloads package a sample can be found for Auto Capture transactions.

SPI Calls

- c. Registration
- d. Finalization
- e. Capture / Reversal (required only in case Manual Capture has been configured)

Please refer to section 5.2 *SPI Calls* for any help and documentation related to above mentioned SPI calls.

Note: Please strictly follow the calling sequence.

3.3. 3D-Secure Model

In compliance with 3D-Secure security guidelines, IPG provides a common payment portal, which can be used by the merchant, for redirecting the card-holder once he is ready to provide the credit card details. The information is then taken by IPG, through the Merchant Plug-in and passed to the Issuer for Authentication.

A Merchant logo can be used on this payment portal to identify the merchant. The Merchant can modify design (fonts, color scheme) of the Payment Portal to make it appear to be a part of the merchant's website.

3D Secure is the most secure payment method available for payment. In addition to the steps mentioned in *Redirection Model* after Card validation on CPP IPG looks for 3D Secure URL provided by the card issuer bank and opens the bank's 3D Secure URL in a pop-up. This page has been hosted by the bank itself and buyer provides the information like PIN or Password that's already been communicated to the buyer by his bank. If authenticated, response is given back to CPP and CPP redirects back to Merchant's redirection link for Finalization and other steps to complete the payment.

Figure 9 shows the business workflow of a transaction following Redirection Model.

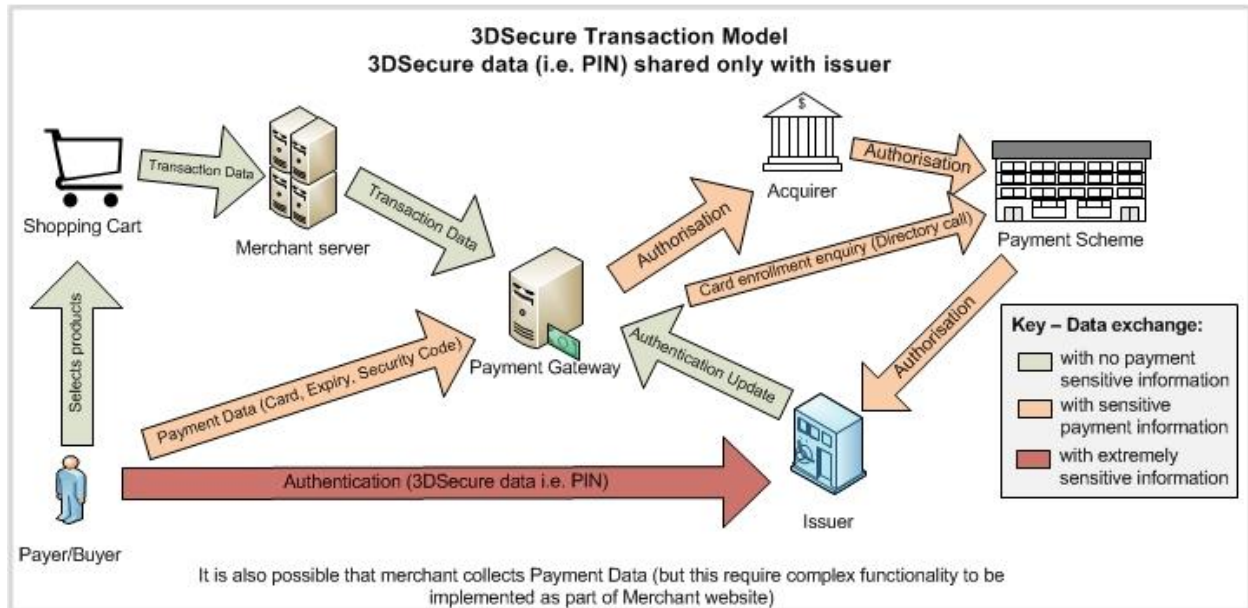


Figure 9

Implementation

For the implementation of 3D Secure, no settings or configurations have to be done at Merchant's side. Merchant bank specifies in Work Order regarding if 3D Secure is applicable or not and if it is, then configurations have to be done on IPG end during the provisioning process.

3.4. Recurring Payments Model

Recurring Payments is a solution where a Merchant wants to save Credit Card information (sensitive data) and payer gives him permission to make future transactions on his behalf or may be on just one click by payer (payer does not need to provide same Credit Card information again and again) for his future transactions.

As per PCI standards, only PCI compliant companies are allowed to store any Credit Card sensitive data like card number. Every merchant who wants to implement Recurring Payments kind of functionality cannot afford to be PCI compliance.

So IPG is providing "Recurring Payments" feature where merchants will redirect the payers to IPG where they'll provide their Credit Card details, IPG will authenticate provided data and return a reference for saved Card details for future Recurring Payment call from same Merchant for that specific Credit Card.

Implementation:

Implementation of Recurring Payments is a **two-step** process:

3.4.1 Step-1- Registering Credit Card for recurring payments

To register a Credit Card for recurring payments, a normal SPI Registration call is sent using any implementation of Redirection Model (Section 3.1). Following parameters needs to be added to identify that call is treated for registering a Credit Card:

Parameter	Value
Recurrence/Type	M

Following things must be noted or considered once SPI Registration call is identified as a recurring payment call:

- None of the SPI calls taking part in Redirection Model (Section 3.1) life cycle will be changed in any way except for the above mentioned change SPI Registration call.
- **TransactionID** returned in SPI Finalization call will be treated as **RecurrenceID** to be used in future recurring payment calls (this is the reference merchant needs to store to point to card details just saved for recurring payments).
- Returned TransactionID will also be treated as unique reference for current payment.
- This call will redirect the payer to IPG Common Payment Page where payer will be prompted to provide Credit Card information. On successful authentications (including verification code and 3D Secure if configured) the amount mentioned will be processed and Credit Card data will be saved for future reference.

3.4.2 Step-2- Recurring Payment through Authorization Call or Registration Call

In step two there are following two options to do the recurring payment.

3.4.2.1 Option-1- Recurring Payment through Registration Call for a registered Credit Card

To make payments against an already registered Credit Card for recurring payments, SPI Registration call is sent using implementation of Registration Model (Section 3.2) with following changes:

- In the registration call RecurrenceID returned by SPI Finalization call in STEP I (Section 3.4.1) will be provided in the ExtraData property of registration call with the name RecurrenceID and the value of the RecurrenceID. Refer to section 5.3.7 for more information on ExtraData.

Following is an example how the RecurrenceID can be set in the registration call.

```
pay.SetProperty ("ExtraData/ RecurrenceID ", "Put here TransactionID of 'Original Registration Transaction' ");
```

Following things must be noted or considered once SPI Registration call is identified as a recurring payment call:

- Each call will automatically get and use the Credit Card details saved during registration in step 3.4.1.
- Once Payer is redirected to the payment portal Payer will not be required to provide the credit card details other static information will be displayed on the payment portal like amount, currency etc.
- If the Credit Card is 3d secure payer will be requested to provide the 3d secure information. Once authorized successfully payer will continue with the payment and will be redirected to the provided merchant URL from where the merchant can initiate the finalization call.
- This subscription for recurring payments will stay valid till Credit Card expiry date.
- Any details for a registered Credit Card cannot be changed.
- Payer will be asked to enter CVV for 'subsequent recurring 3D transactions'
- A new and unique TransactionID will be returned as a reference against each SPI Registration call (please don't confuse and/or replace 'Subsequent Recurring Transaction' with the 'Original Registration Transaction').

Note: This is a 3d secure model and payer card will be authenticated each time the recurring payment is being made.

Disclaimer: Please note that for Recurring 3D transactions, business rules that involve fees or discounts can't be used. Such business rule hint will fail the transaction flow.

3.4.2.2 Option-2- Recurring Payment through Authorization Call for a registered Credit Card

To make payments against an already registered Credit Card for recurring payments, SPI Authorization call is sent using any implementation of Authorization Model (Section 3.1) with following changes:

- Credit Card number should be skipped and not mentioned
- Credit Card expiry fields should be skipped and not mentioned
- Verification Code should be skipped and not mentioned

Please note 'according to PCI customers can't store cards, BUT they can COLLECT it, without storing it'. Following parameters needs to be added to identify that call is treated for already registered recurring payment:

Parameter	Value
TransactionID	RecurrenceID returned by SPI Finalization call in STEP I (Section 3.4.1)

Following things must be noted or considered once SPI Authorization call is identified as a recurring payment call:

- Each call will automatically get and use the Credit Card details saved during registration.
- Payer will not be prompted again for any authentication.
- This subscription for recurring payments will stay valid till Credit Card expiry date.
- Any details for a registered Credit Card cannot be changed.
- A new and unique TransactionID will be returned as a reference against each SPI Authorization call (please don't confuse and/or replace 'Subsequent Recurring Transaction' with the 'Original Registration Transaction').

NOTE: This is a non-3D mode of recurring payments, even if the 'Original' transaction is 3D secure, this doesn't mean merchant is protected for the subsequent transaction. Each transaction is a separate transaction in itself, and therefore liability applies individually'.

3.5. Comparison between Manual Capture and Auto Capture

Auto Capture	Manual Capture
Technical Differences	
In SPI Registration call, TransactionHint property has a value 'CPT:Y' along with other ';' separated values.	In SPI Registration call, TransactionHint property has a value 'CPT:N' along with other ';' separated values.
Requires no extra call or interaction.	Requires to manually initiate a capture/reversal call (may or may not involve user interaction, depending on business requirements).

Full amount mentioned to be paid will automatically be captured instantly after blocking without any control to stop this action (Capture seems to be a part of SPI Finalization call).	In this case SPI Finalization will get the response back only blocking the amount.
Allows to only capture full amount automatically in one go.	Allows both SPI: Capture and SPI: Reversal calls. If no amount and currency has been mentioned, it processes full amount available as blocked amount which is a default behavior. If specific amount and currency has been provided for capture or reversal, only mentioned amount is processed (provided that it does not exceed amount available as blocked). This is called Partial Capture/Reversal.
Only single capture request.	Multiple capture/reversal calls can be sent at different times before all blocked amount is captured/reversed.
SPI: Finalization call closes the transaction itself.	Transaction is not closed at SPI: Finalization instead it's closed when full amount has been captured or reversed.
Business Differences	
Auto Capture is recommended if your business flow does not need any physical/logical authentication/validation, for example, a retail store.	If business requires any physical/logical authentication/validation like in case you have to validate a physical delivery address or amount has to be captured subject to goods received.
Done seamlessly and automatically.	Sometimes need extra manual effort and delegation of a user action if the authentication/validation is not automatic in Merchant application.
IPG closes the transaction itself.	Merchant is responsible for any captures/reversals or what so ever.

4. IPG INTEGRATION GUIDE

IPG provides the merchant with an ePayment Platform Integration tool, called Store Payment Interface - **SPI**. This will enable the merchant to integrate their website / store easily with the payment platform. The SPI is a product which sits on the Merchant Server and controls communication between the IPG ePayment Platform and the Merchant Server.

Before doing the final technical level configurations and integration, there several post requisites to be considered and answered which are stated as under.

4.1. Payment Specific Decision

4.1.1. Web-based payments decisions

For web based payments the merchant can select one of three modes. Each mode consists of various features being available, different security requirements and integration processes. Redirection Model (explained above in detail) applies to these kinds of transactions.

4.1.1.1. *Integrated Payment Portal*

The most common way for processing web-based payment is through the Integrated Payment Portal. Merchants using this mode are not required to collect any payment information on their own. After receiving the payer's confirmation of purchase intent, the merchant will calculate the value of goods and services and register the transaction with the IPG. If the registration has been processed successfully, the IPG will return transaction identifier (TransactionID) and URL for Payment Page. This identifier can be used for subsequent tracing of the order and the URL of the Payment Portal, which in turn is used by the merchant for redirection. After redirection, the payer will be presented with the IPG Common Payment Portal where he will provide payment details and be guided through the authentication process (if required). On completion, the payer will be redirected to the merchant's website and the merchant will decide if they want to go ahead with transaction. If so, they can finalize the transaction which will result in either blocking payer funds against transaction requested or funds transferred directly to the merchant's account (both can take place only if finalization was successful).

While the Payment Portal is used by multiple merchants, it allows each merchant to customize the look and feel in several ways. The simplest one by default contains the merchants' beneficiary details (including merchant name, city and country). The Merchant can also add their own logo, place the Payment Portal within the frame of their website and even create their own style sheet.

4.1.1.2. *Merchant payment entry page using authorization API*

In certain cases the merchant may want to have full control over the input of card information by payer - they may have certain rules which cannot be easily translated into the form of scenarios, or wish to make payment information entry an integral part of their website. In these cases, the merchant can collect payment details on their side through the page developed by their team. The merchant payment information entry page has great value for the merchants that want to maintain a consistent look and feel through the whole process, which is not always possible even when using style sheets. Further benefits include allowing the merchant to take multiple decisions (and even include the whole workflow) from the moment the payment data was provided to processing the transaction through IPG. However, acceptance of payment data has its own cost as the merchant is subject to PCI DSS rules regarding storage of sensitive data

and processes around it. The merchant application should preferably possess knowledge of card ranges belonging to particular brands whenever extra security information should be asked for. Additionally the merchant will not be able to perform BizDirect nor eDirham payments and their transactions will not be processed in the 3DSecure compliant fashion thus exposing them to liability for any fraud made. From a technical point of view, the merchant will use a single Authorization call instead of Registration, redirection and Finalization calls present in the Payment Portal enabled implementation.

4.1.1.3. SPILess Payment

In certain situations where the merchant wants to process payments, their web server does not allow installing SPI components or the appropriate SPI since such a platform does not exist. In these cases merchants may opt to choose the SPILess payment approach unlike those described earlier. This situation is most common in shared hosting environments where the server owner allows only components approved by them to be present on the system. Similar situation takes place if outgoing access is not allowed from the system. Since most of the SPI versions have to be installed or configured by system administrator and all require outgoing access (to connect to IPG), such situations make standard processing impossible.

To allow these merchants to transact online (rather than process requests in MOTO mode) an additional transaction mode is possible. In this mode the merchant rather than sending transaction details to the IPG using SPI component, simply posts them (as HTTP Post hidden fields) to the SPILess Portal. The portal takes data and internally registers them with IPG before following standard process through Payment Portal. At the end when the Payment Portal would redirect the payer to the merchant's web page for Finalization, it will redirect the payer to the SPILess payments which will do finalization on the merchant's behalf. Finally the payer can be redirected to the merchant's website (if merchant wishes so) or be presented with the completion message.

While the merchant is not expected to install anything on the system, the SPILess payments have certain security drawbacks. This security problem lies in the fact that data between merchant and the SPILess Portal is passed using redirection through the payer browser. Even without advanced knowledge the payer can modify the data sent (i.e. decrease transaction amount, change currency) or modify transaction result.

To avoid fraud related to such cases the merchant is required to manually review the details of each transaction before capturing it and providing service. The SPILess Payments will stop the merchant from attempting to capture transaction in single step to ensure that manual process is done. As a result, payment instruments which require automatic capture (like BizDirect or eDirham) are not available in this mode. Fortunately all other benefits of Payment Portal transaction are kept. If merchant detects any discrepancy they can simply opt not to capture

transaction (and eventually reverse it). Merchants should not rely on the online status of the transaction for any other reason than to notify the operator about the need of a manual review and a general thank you message.

SPILess Payments cannot support merchants selling services which do require immediate delivery and is not recommended for high volume merchants due to the need of manual verification for each transaction. SPILess is the ideal solution for charities (which may use any free hosting offering) and are not exposed to fraud.

4.1.2. MOTO

The merchant has multiple possibilities for processing MOTO transactions. Authorization Model (explained above in detail) applies to these kinds of transactions.

4.1.2.1. *Standard authorization call*

In the MOTO mode the merchant receives payment details directly from the payer over the channel where the Payment Portal does not exist. In these cases, the merchant can enter payment details into their internal application (or even website) and initiate the transaction using the 'Authorization' method. The merchant cannot use the same application as for web based payments just operated by their staff, as using the web payment method indicates that the payment details have been entered by payer himself. The Merchant processing transactions using this method is subject to PCI DSS review both for merchant process and application (as card details are entered onto the merchant's systems).

4.1.2.2. *Customer Activated Terminals (CAT)*

Merchants can accept MOTO transactions from the Customer Activated Terminals (CAT) using the Card Track 2 Data by sending directly in Authorization call. If Track 2 Data is being provided for the transaction then no other Credit Card related information is needed to process the transaction.

4.1.2.3. *Recurring transactions*

In many cases MOTO transactions are 'this transaction of recurring type'. While the merchant could save card details on their system and fire transactions exactly as in standard authorization call, there is a way where merchants can completely avoid PCI DSS implications, benefit partially from 3D Secure liability protection and extend the list of payment instruments (not only credit cards like in two above methods). The merchant would need to request payer to pay the first recurring installment while registering using web payment through the Payment Portal. The only difference is that during registration the merchant should indicate that the registration is for recurring transaction and eventually specify recurrence attributes.



After the first payment is completed (processed in 3DSecure way) the merchant should store the returned TransactionID as the identifier of the recurrence. Subsequently when merchant wants to collect the following payments they should conduct 'authorization', but instead of specifying payment details (which they do not have) they should give the previously obtained TransactionID. If the merchant specified any recurrence attributes, the payment gateway will further verify if the requested transaction does not violate any of them. Using this method imposes certain requirements on merchant process, thus it cannot be used in all cases as registration has to be made through the channel which is supported by Payment Portal. In addition, the first transaction has to be executed during registration, although this could be just a token value transaction reversed in case of success.

4.2. Merchant Specific Decision

4.2.1. Selecting the right capture mode and handling finalization and delivery errors

The merchant should select the capture mode (automatic or delayed) based on the type of service sold. If the merchant is selling physical goods or there is a possibility that the requested service cannot be delivered, the merchant should select delayed capture (TransactionHint CPT:N) and execute the capture separately after fulfillment. N.B. does not include transactions done using BizDirect / eDirham payment methods.

If the merchant decides to use immediate capture (CPT:Y), they should be able to handle situations such as the inability to update their systems or a temporary fulfillment request. In these cases a message should be displayed stating the actual status of the payment transaction. The payer should be informed of the necessary steps to be taken to receive their service or when the service can be delivered. In case the merchant cannot store the transaction status in their system due to a technical problem, the payment gateway TransactionID, ApprovalCode and information regarding how the payer should contact a merchant to complete transaction should be provided. In these circumstances the merchant contacted by a payer should verify its transaction status using the Merchant Portal and either manually complete it or send to the acquiring bank for its refund. In both cases the merchant should keep the payer updated about the status.

There are rare situations when the merchant will receive finalization time-out despite the transaction being completed on both the issuer's and even the payment gateway's end. If such a transaction was sent in CPT:N mode or has been completed only on the issuer side, then the payer will not be charged (although money paid during transaction can be temporary blocked).

However, if the transaction reached the payment gateway and automatic capture got executed, the payer can be charged. Both situations should be resolved during reconciliation - when the merchant compare its records with the payment gateway records. Furthermore, the merchant should execute Reversal (in case of CPT:N done from Merchant Portal) or Refund directly through the acquirer or BizDirect bank. Eventually if merchant maintains the payer details, they can contact the payer and make a joint decision regarding when to deliver service or reverse / refund payment. All situations where the merchants were unable to fulfill their part of transaction while the payer was either charged or money on his account has been blocked are referred to as technical failures (regardless of whether it occurred on the merchant side or somewhere else).

BizDirect and eDirham transactions cannot be process in delayed capture mode, thus using CPT:N will exclude such payment mechanisms from list of available tokens for the payer. On the other hand, delayed capture should be used for credit card transactions where the delivery of physical

goods is involved or even in the case where a service is being purchased. It would significantly simplify the actions to be taken by the merchant in the case of any technical failure.

4.2.2 SPI Platform Selection

The SPI is available in several flavors to suit the platform of choice of the merchant. The SPI also has built-in intelligence to take controlled decisions and logic on the information it services. The SPI is usually setup with all the parameters for the merchant, including the Merchant-ID.

Current versions of SPI include:

a. SPInet

Windows .net Component for ASP.net and .net standalone applications (may also work on non-Windows platforms where flavors of .Net framework exists).

b. SPIj

Any platform running Java. Java application servers and java applications (java can be also run inside of non-java based servers i.e. PHP).

c. SPI Windows Service

SPIs is a windows service that allows IPG Merchants to communicate to IPG using simple SPI calls.

d. SPIless

No SPI installed on the merchant server. HTML page redirected to a hosted page. Automated capture not possible with this option - Capture can be done manually through a Web User Interface.

4.2.3. Enabling Verification Code

By default Card Verification Code is not required to make a transaction. If merchant wants it to be required, a property called **TransactionHint** has to be configured during SPI Registration call. Please refer to section 5.3.4 (Transaction Properties) for details on TransactionHint.

4.2.4. Enabling 3D Secure Authentication

For accepting payments using 3D secure, merchant needs to contact their bank, bank provides the details for 3D secure in the Work Order to IPG. There are no specific configurations needed on merchant side to process payments using 3D secure authentication.

5. SPI USER GUIDE

This section points to SPI installation, configuration and troubleshooting guide with API for SPI calls. SPI along with samples, configuration tools and test digital certificates can be downloaded from the following link.

Download Link:

<https://demo-ipg.comtrust.ae/merchant/#/downloads>.

5.1. SPI Installation Guide

5.1.1. Requirements

- Every version of SPI requires specific installation and configuration steps, but all share some of system level configurations and tests.
- It is assumed that runtime environment i.e. Java for SPIj and .net framework for SPI.net has already been installed and configured on Merchant's server machine. This guide does not cover any steps related to such configurations.

5.1.1.1. Firewall/Router Configuration

The following outbound connections have to be allowed from the server hosting SPI:

- **IPG Production:** ipg.comtrust.ae [195.229.85.91]:2443 [SSL]
- **IPG Staging:** demo-ipg.comtrust.ae [195.229.84.29]:2443 [SSL]

5.1.1.2. Connectivity Testing

To test the connectivity and router/firewall setting applied in the previous step please use the following lines:

- **IPG Production:** telnet ipg.comtrust.ae 2443
- **IPG Staging:** telnet demo-ipg.comtrust.ae 2443

The connection on ports 2443 is based on SSL and as such is not fully understood by simple system applications as telnet, however it allows confirming the basic connectivity (telnet should connect to the port without showing any kind of error like for example connection refused or listener not found).

NOTE: IPG refuses any connections coming from proxy.

5.1.1.3. Digital Certificate

For secure communication with IPG, SPI needs to establish an SSL connection to IPG server using Digital Certificate issued by Etisalat. Merchant can apply for a Digital Certificate (Business User Certificate) at <https://comtrust.etisalat.ae/enrollment/app/general/DirBusinessUser>.

It is highly recommended that when you get your certificate from PKI you perform the following procedure:

Make sure that certificate is in proper format; it should be password protected by importing your certificate into any windows system and export it to get two copies using default properties:

- With private key (never to be shared with anyone even IPG team)
- With private key and without certificate chain (for SPIj)
- Without private key

Quick way to import/install the certificate in windows based machine is; double click the pfx file and follow the instructions through the wizard carefully; keeping above mentioned 2 points in mind. For instructions on importing or exporting Digital Certificates, please refer to [http://technet.microsoft.com/en-us/library/cc782788\(W5.10\).aspx](http://technet.microsoft.com/en-us/library/cc782788(W5.10).aspx).

Note: Certificate cannot be sent again. If lost, you have to apply for a new certificate.

For testing, test certificate for test merchant ‘Demo Merchant’ can be downloaded from downloads link provided above in section 5. Password for all Demo Merchant certificates is **Comtrust**.

5.1.2. SPInet

For windows based server environments, our recommended flavor of SPI is SPInet. SPInet has been built in Microsoft .NET Framework. SPInet versions for both .NET Framework 2 and .NET Framework 3 or higher can be downloaded from the download link provided in section 5. We highly recommend using SPInet for .NET Framework 3 or higher.

5.1.2.1. System Requirements

- Any machine capable of running Microsoft .NET Framework
- Microsoft .NET Framework
- TCP connectivity to Internet Payment Gateway (refer to section 5.1.1)

5.1.2.2. Installation

1. Copy SPI.dll provided in SPInet download package into your system
2. Add reference to copied SPI.dll into your project

5.1.2.1. SSL Requirements

In order to be able to establish a secure connection to IPG, merchant has to poses a client certificate (refer to section 5.1.1.3).

To install the certificate see the instructions mentioned in section 5.1.1.3.

5.1.2.2. Configuration

Run SPIConfig.exe provided under “Configuration Utility” folder in specific SPInet download package. For initial testing and configuration, use the following configurations:

Property	Value (Bold value is for sample for testing)
Customer	Demo Merchant – This is the customer id as provided by IPG in Work Order

Channel	(leave it blank) – Payment channel through which payments has to be made, if not provided; default is Web
Store	(leave it blank) – May or may not be provisioned to your account as per Work Order
Terminal	(leave it blank) – May or may not be provisioned to your account as per Work Order
Address	demo-ipg.comtrust.ae – IPG Server address
Port	2443 – Communication Port
Timeout	120 – Timeout for request in seconds
Secure SSL	True
Certificate	Path of Demo Merchant certificate file having private key (.pfx file) refer to section 5.1.1.3 – Certificate file path for the certificate provided by PKI
Password	Comtrust – Password for the client certificate file (.pfx file) provided at the time of exporting certificate with private key (refer to section 5.1.1.3)
Card Number	999000000000029, 999000000000003, 999000000000011 (Test Cards)
Currency	AED
Expiry Month	Any – Credit Card expiry month
Expiry Year	Any – Credit Card expiry year

Providing these configuration properties and clicking the Test button will fire a transaction and if Response Code returned is 0 it means configurations are fine for Demo Merchant customer. To test the configuration for your user, provide the properties w.r.t. your user (from Work Order) and hit Test button to fire a transaction from your account. To resolve the issues (if faced) see the document mentioned in section 1.1 to get the Response Code 0.

SPI net configurations can be done in two ways:

5.1.2.2.1. Loading from configurations file

Once Configuration Utility returns Response Code 0; click “Create Web Config” button and copy the settings into Web.config file for your project. Sample code can be found in SPI net download package.

5.1.2.2.2. Manual configurations

Once Configuration Utility returns Response Code 0; you have to manually provide connection settings and configuration data into the code.

For manual configurations initialize the **Transaction** object as provided below and use sample code for setting up transaction properties.

```
Transaction pay = new Transaction (false);
pay.Initialize("Registration", "2.0");
// Merchant ID from Work Order
pay.Customer = "Demo Merchant";
// Initiaizing the connection
pay.Connection = new Comtrust.Payment.IPG.SPInet.Connection();
// IPG server address
pay.Connection.Address = "demo-ipg.comtrust.ae";
// Certificate file path for .pfx file
pay.Connection.CertificatePath = @"C: \Demo Merchant.pfx";
// Certificate password
pay.Connection.CertificatePassword = "Comtrust";
// Connection port
pay.Connection.Port = 2443;
// Securing the connection for communication
pay.Connection.IsSecure = true;
// Timeout for connection
pay.Connection.TimeOut = 120;
// Payment channel
pay.Channel = "Web";
// Language code
pay.Language = "en";
// Store - do not mention if default store has to be hit
//pay.Store = "Store Name";
// Store - do not mention if default terminal has to be hit
//pay.Terminal = "Terminal Name";
pay.Execute();
```

5.1.3. SPIj

This section covers SPI implementation for Java based server environments. For **PHP** based servers, [PHP/Java Bridge](#) can be used (IPG does not provide any support for PHP/Java Bridge or related issues, Merchant has to do this on his own). Please download SPIj package available at download link provided in section 5.

5.1.3.1. System Requirements

- Any machine and operating system capable of running java (the ability to run graphics applications is recommended but not required)

- TCP connectivity to Internet Payment Gateway (refer to section 5.1.1)
- Java2 Runtime Environment 1.4.x or later (no additional components required)

5.1.3.2. Installation

- Download SPIj package from the Download link provided in section 5
- Unpack zip file to the directory where you plan to have SPIj installed
- Add SPI.jar file to your java classpath

5.1.3.3. SSL Requirements

In order to be able to establish a secure connection to IPG, merchant has to poses a client certificate (refer to section 5.1.1.3) which is trusted by IPG and allows SPIj to trust the certification authority which issued IPG server certificate. The certificates are kept in two different stores *SPIMerchant* store and *SPIMTrust* store. The first one is client certificate while the second one contains the list of trusted authorities.

Client certificate can be acquired by following the steps mentioned in section 5.1.1.3 while *SPIMTrust* is available in SPIj download package, in *Certificates* folder password for *SPIMTrust* store is **password**.

5.1.3.4. Configuration

1. Open SPI.properties file
2. Update MerchantKeystore property to reflect the location of your certificate (.pfx file provided by PKI). Pfx file must not include certificate chain (refer to section 5.1.1.3)
3. Update MerchantKeystorePassword with the password to your certificate
4. Update TrustedKeystore property to reflect the location of SPIMTrust file
5. Make sure that certificate is in proper format, it should be password protected and without CA chain (to make sure that the format is as described above please import your certificate into any windows system and then export it using following options:
 - With private key
 - Without certificate chain
 - Without any additional properties or higher encryption schemes
 - With password
6. In SPIj
7. Run the test tool provided to make sure the configurations are working fine
8. Sample and test projects/files can also be found in SPIj download package

5.1.4. SPIs (Windows Service)

SPIs is a windows service that allows IPG Merchants to communicate to IPG using simple SPI calls. This version of SPI is useful especially in cases where implementation of SPI.net and SPIj is not an option like in cases of legacy applications using COM Components.

5.1.4.1. *System Requirements*

- Any machine and operating system capable of Microsoft .Net framework 4 or later
- TCP connectivity to Internet Payment Gateway (refer to section 5.1.1)

5.1.4.2. *Installation*

- Download SPIs package from the Download link provided in section 5
- Run setup.msi file available in the package
- Choose default location for installation
- After installation is completed, go to installation directory “[Program Files]\Comtrust\Integrated Payment Gateway\SPI Service\” (if default installation is chosen)
- Run SPIConfig.exe

5.1.4.3. *SSL Requirements*

In order to be able to establish a secure connection to IPG, merchant has to poses a client certificate (refer to section 5.1.1.3) which is trusted by IPG and allows SPIj to trust the certification authority which issued IPG server certificate. The certificates are kept in two different stores *SPIMerchant* store and *SPIMTrust* store. The first one is client certificate while the second one contains the list of trusted authorities.

Client certificate can be acquired by following the steps mentioned in section 5.1.1.3.

5.1.4.4. *Configuration*

Refer to SPIs Integration Guide provided in SPIs package.

5.1.5. SPILess

SPILess is usually used whenever using SPI is not an option, however you’ll be very limited in what you can do with it and it’ll required more manual effort from your side.

The payment gateway uses the combination of the Customer ID and his digital certificate (provided by Etisalat) to authenticate the request but this cannot not be done with SPILess, Since

you as the merchant cannot be authenticated you'll need to send the required information (check the sample below) and therefore you can only block the amount of money from the credit card.

To actually transfer the money into your account you'll be required to login using the digital certificate to our Merchant Web Interface and capture the money manually.

As you can see in the sample below, the request is just a simple HTML form that can be sent by anyone so the person using the Merchant Web Interface will have to manually check the records on your database and link them to the transactions in the web interface before capturing them.

Below is the SPILess sample (with the minimum required parameters, you can add more parameters such as Order ID and Name etc. mentioned in SPI user guide at below downloads page). You can use it to test how SPILess works (use the IPG Card 999000000000029) and you can install the Demo Merchant certificate (available in downloads page at <https://ipg.comtrust.ae/merchant/#/Downloads>) on your machine. Then you can login to our staging Merchant Web Interface <https://demo-ipg.comtrust.ae/merchant/> (in staging you'll have to select Demo Merchant from the Customer List) and check your test transactions (use the Not Captured report to find transactions with block amount only, also note that many other customers are testing using Demo Merchant on our staging system so you'll find transactions not related to you).

```
<html>
<body>
  <form method="POST" action="https://demo-ipg.comtrust.ae/SPIless/Registration.aspx">
    <table>
      <tr><td>Amount:</td><td><input type="text" name="Amount" value="99.98"/></td></tr>
      <tr><td>Currency:</td><td><input type="text" name="Currency"
value="AED"/></td></tr>
      <tr><td>Customer:</td><td><input type="text" name="Customer" value="Demo
Merchant"/></td></tr>
      <tr><td>Language:</td><td><input type="text" name="Language" value="en"/></td></tr>
      <tr><td colspan="2"><input type="submit" value="Pay"/></td></tr>
    </table>
  </form>
</body>
</html>
```

5.2. SPI Calls

In reference to Payment Models (refer to section 3), following are the details for IPG calls that a merchant can make using SPI.

Please find the detailed usage and documentation of properties used in SPI Calls in following section i.e. section 5.3.

NOTE: Following sections are common and mandatory inputs/parameters to every SPI call other than the ones mentioned inputs/parameters of SPI call specific properties:

1. Connection Properties (Section 5.3.1)
2. SPI Specific Connection Properties (Section 5.3.2)

5.2.1. Registration

This is the first call in Redirection Model (refer to section 3.2). Merchant is required to provide all the details for the transaction including following mandatory and optional list of properties (this list does not include configuration properties/settings mentioned in sub sections of 5.1):

Property	Usage	Comments
Request Properties		
Customer	Customer ID. Please refer to section 5.3.3 for Customer details	MANDATORY
Store	Store. Please refer to section 5.3.3 for Store details	OPTIONAL
Terminal	Terminal. Please refer to section 5.3.3 for Terminal details	OPTIONAL
Channel	Payment Channel. Please refer to section 5.3.3 for Channel details.	MANDATORY
Amount	Total amount to be charged.	MANDATORY
Currency	Currency in which above mentioned amount is to be charged.	MANDATORY
OrderID	Merchant can use this property to map id for this transaction w.r.t. his system (can also be auto generated, please refer to section 5.3.4 for details).	OPTIONAL
ReturnPath	Specifies the URL a buyer will be redirected back to in case Redirection Model (refer to section 3.2) has been adapted.	MANDATORY
OrderName	Short description for order.	MANDATORY

OrderInfo	Details of order.	OPTIONAL
TransactionHint	It is used to specify which payment instruments should be available to the buyer. Merchant can specify which features should be supported by payment instrument i.e. Auto Capture/Reversal or Manual Capture/Reversal. By default it is set Auto Capture. please refer to section 5.3.4 Transaction Properties for details	OPTIONAL
Recurrence/Type	Marks a transaction for registering Recurring Payments.	MANDATORY for registering Recurring Payment
Response Properties		
TransactionID	Reference for ongoing transaction to be used in all SPI calls made for this transaction. TransactionID should be saved by Merchant for any future references to a particular transaction in IPG system.	
PaymentPortal	Link to IPG Common Payment Page (CPP) where payer has to be redirected for providing Credit Card information for next step in completing the transaction.	

5.2.2. Finalization

Once payer has been redirected back to Merchant website (ReturnPath provided in Registration) from CPP after providing Credit Card information, Merchant has to send SPI Finalization call to actually block the money from payer's card in case of Manual Capture and to complete the transaction in case of Auto Capture. Following is the list of properties applicable for SPI Finalization call (this list does not include configuration properties/settings mentioned in sub sections of 5.1):

Property	Usage	Comments
Request Properties		
Customer	Customer ID. Please refer to section 5.3.3 for Customer details	MANDATORY
Store	Store. Please refer to section 5.3.3 for Store details	OPTIONAL
Terminal	Terminal. Please refer to section 5.3.3 for Terminal details	OPTIONAL

Channel	Payment Channel. Please refer to section 5.3.3 for Channel details.	MANDATORY
TransactionID	TransactionID returned in SPI Registration call.	MANDATORY
Response Properties		
OrderID	It's returned only in case of successful transaction and if it had been set during Registration call for same	
ApprovalCode	ApprovalCode, as sent by the issuer bank. Merchant should save this code for future reference and communication with issuer bank related to a particular transaction.	
Amount	Amount charged for the transaction	
Balance	Balance amount for the transaction that has not yet been captured	
CardNumber	Masked credit card number in following format: xxxxxx*****xxxx It will return first 6 and last 4 digits of credit card used for payment.	
CardToken	Tokenized value for the card used, it's not original credit card number but will always be same for any particular credit card number	
Account	Name of account in payment gateway configurations that transaction happened with (to avoid any confusions, use this field for any references or logging only if advised by Merchant Integration Support)	

5.2.3. Authorization

Authorization Model (refer to section 3.1) includes only one call (if Auto Capture has not been enabled; whenever Manual Capture has been mentioned by the Merchant, Capture/Reversal call is mandatory to complete and close a transaction else capture is done automatically). Following are the set of properties used in different modes (refer to section 3.1) of this SPI Authorization call.

Property	Usage	Comments
Request Properties		
Customer	Customer ID. Please refer to section 5.3.3 for Customer details	MANDATORY
Store	Store. Please refer to section 5.3.3 for Store details	OPTIONAL

Terminal	Terminal. Please refer to section 5.3.3 for Terminal details	OPTIONAL
Channel	Payment Channel. Please refer to section 5.3.3 for Channel details.	MANDATORY
Amount	Total amount to be charged.	MANDATORY
Currency	Currency in which above mentioned amount is to be charged.	MANDATORY
CardNumber	Credit Card number	MANDATORY (1)
ExpiryYear	Expiry year of Credit Card (please refer to section 5.3.5 for format)	MANDATORY (1)
ExpiryMonth	Expiry month of Credit Card (please refer to section 5.3.5 for format)	MANDATORY (1)
VerifyCode	Credit Card Verification Code	MANDATORY (1)
CardTrack2	Card Track2Data (read from card magnetic tape or chip like in case of kiosk devices)	MANDATORY (2)
OrderID	Merchant can use this property to map id for this transaction w.r.t. his system (can also be auto generated, please refer to section 5.3.4 for format).	OPTIONAL
OrderName	Short description for order.	OPTIONAL
OrderInfo	Details of order.	OPTIONAL
TransactionHint	It is used to specify which payment instruments should be available to the buyer. Merchant can specify which features should be supported by payment instrument i.e. Auto Capture/Reversal or Manual Capture/Reversal. By default it is set Auto Capture. please refer to section 5.3.4 Transaction Properties for details	OPTIONAL
TransactionID	RecurrenceID for a registered for a Credit Card recurring payments.	MANDATORY (3) for Recurring Payment
Response Properties		
TransactionID	Reference for ongoing transaction to be used in all SPI calls made for this transaction. TransactionID should be saved by Merchant for any future references to a particular transaction in IPG system.	
ApprovalCode	ApprovalCode, as sent by the issuer bank. Merchant should save this code for future reference and communication with issuer bank related to a particular transaction.	

OrderID	It's returned if it's set to be auto generated.	
---------	---	--

NOTE:

1. Please make sure to send either **CardNumber and SecureCode** or only **CardTrack2**
2. Please make sure to send either **ExpiryYear and ExpiryMonth**
3. For Recurring Payment call, do not provide **CardNumber, SecureCode, ExpiryYear, SecureCode or CardTrack2**.

5.2.4. Capture

For any transaction in IPG whether it is following Redirection Model (refer to section 3.2) or Authorization Model (refer to section 3.1), a transaction can be marked to be captured automatically or manually. For manual capture if amount has to be transferred from payer to merchant, SPI **Capture** call is required. Capture has two modes

- Complete Capture
Total outstanding balance amount is captured.
- Partial Capture
Portion of outstanding balance amount is captured.

Following are the properties used in SPI Capture call for both modes:

Property	Usage	Comments
Request Properties		
Customer	Customer ID. Please refer to section 5.3.3 for Customer details	MANDATORY
Store	Store. Please refer to section 5.3.3 for Store details	OPTIONAL
Terminal	Terminal. Please refer to section 5.3.3 for Terminal details	OPTIONAL
Channel	Payment Channel. Please refer to section 5.3.3 for Channel details.	MANDATORY
TransactionID	TransactionID returned in SPI Registration or Authorization call.	MANDATORY
Amount	Amount to be captured. This Amount should be less than or equal to outstanding balance amount.	MANDATORY for Partial Capture
Currency	Currency in which above mentioned amount is to be charged. This Currency should be same as that of mentioned in SPI Registration or Authorization call.	MANDATORY for Partial Capture

TransactionHint	Only allowed value is RVS:Y or RVS:N which indicates whether remaining part of balance should be reversed automatically or not.	OPTIONAL
Response Properties		
TransactionID	Reference for ongoing transaction.	
Balance	Outstanding balance after current transaction.	

5.2.5. Reversal

For any transaction in IPG whether it is following Redirection Model (refer to section 3.2) or Authorization Model (refer to section 3.1), a transaction can be marked to be captured automatically or manually. For manual capture if amount has to be unblocked for not to be transferred from payer to merchant, SPI **Reversal** call is required. Reversal has two modes

- Complete Reversal
Total outstanding balance amount is reversed / unblocked.
- Partial Reversal
Portion of outstanding balance amount is reversed / unblocked.

Following are the properties used in SPI Capture call for both modes:

Property	Usage	Comments
Request Properties		
Customer	Customer ID. Please refer to section 5.3.3 for Customer details	MANDATORY
Store	Store. Please refer to section 5.3.3 for Store details	OPTIONAL
Terminal	Terminal. Please refer to section 5.3.3 for Terminal details	OPTIONAL
Channel	Payment Channel. Please refer to section 5.3.3 for Channel details.	MANDATORY
TransactionID	TransactionID returned in SPI Registration or Authorization call.	MANDATORY
Amount	Amount to be reversed / unblocked. This Amount should be less than or equal to outstanding balance amount.	MANDATORY for Partial Reversal
Currency	Currency in which above mentioned amount is to be charged. This Currency should be same as that of mentioned in SPI Registration or Authorization call.	MANDATORY for Partial Reversal

Response Properties		
TransactionID	Reference for ongoing transaction.	
Balance	Outstanding balance after current transaction.	

NOTE: If outstanding balance is greater than zero, multiple **Capture** and/or **Reversal** calls can be posted to IPG in order to finally get outstanding balance as zero.

5.3. SPI Transaction Properties

In reference to SPI Calls (please refer to section 5.2) this section deals with the details, valid values and any other significant information related to the properties required or returned by SPI.

5.3.1. Connection Properties

Property	Description
ConnectionAddress	DNS name of Payment Gateway <i>Note: Due to the way SSL connections work, it is not possible to establish a secure connection to the IP address in a form of xx.xx.xx.xx (e.g. 195.229.85.91), in order to establish a connection DNS name must be specified i.e. ipg.comtrust.ae for IPG Production and demo-ipg.comtrust.ae for IPG Staging servers.</i>
ConnectionPort	The port SPI should connect to should be 2443 for SSL connections
ConnectionTimeout	Connection timeout in seconds (120-300) <i>Note: this value is recommended to be 120 or more, any value below 120 may result in transaction inconsistency on the slow network or in heavy load conditions.</i>
ConnectionSecure	(true/false) specifies if SSL or non SSL type of connection is used, IPG does not support non SSL connection for SPI calls. So this property should always be set to true .

5.3.2. SPI Specific Connection Properties

Property	Description	Targeted SPI
MerchantKeyStore	Name of the <i>keystore</i> where the merchant certificate is stored (physical directory path to client user certificate)	SPIj
MerchantKeystorePassword	Password of the <i>keystore</i> where the merchant certificate is stored (password for pfx file)	SPIj
TrustedKeystore	Name of the <i>keystore</i> which contains certificates of certification authorities which should be trusted by SPI (this data will be used to validate Payment Gateway server certificate). This file is same as found in SPIj	SPIj

	downloads package in samples (SPIMTrust), just mention the physical directory path to this file	
TrustedKeystorePassword	Password of the <i>keystore</i> which contains certificates of certification authorities. As file (SPIMTrust) is to be copied from SPIj downloads package, password for that file is password	SPIj

5.3.3. Point of Sale Properties

Property	Description
Customer	This property maps to Customer ID as mentioned in Work Order, for testing on staging Demo Merchant should be used as Customer ID.
Channel	Payment Channel to be used for the transaction <u>Acceptable Values:</u> • Web (default) • Terminal • POS • Recurring • Phone • Mail • USSD
Store	The name of the store (this property is optional unless the merchant requested support for more than one store or the default store has not been provisioned in Payment Gateway, <i>Merchant Integration Support team advises the merchant on its usage upon request</i>)
Terminal	The name of the terminal (this property is optional unless the merchant requested support for more than one terminal or the default terminal has not been provisioned in Payment Gateway, <i>Merchant Integration Support team advises the merchant on its usage upon request</i>)

5.3.4. Transaction Properties

Property	Description
Language	This property can be used with any request and it specifies which language is used for error message descriptions. In order to display messages correctly, the proper language code page has to be installed on the server. Currently defined languages: - EN – English - AR – Arabic

	- QQ – Technical descriptions (detailed description suited for troubleshooting and testing, but not recommended to be used as an end user messages)
Amount	It represents the transaction amount (in standard format with dot as a separator e.g. 12.34)
Currency	Transaction's currency as ISO currency code (e.g. 840 for US Dollar, 874 for UAE Dirham) or ISO currency name (e.g. USD for US Dollar, AED for UAE Dirham). Please refer to following link for complete list: http://www.currency-iso.org/iso_index/iso_tables/iso_tables_a1.htm
TransactionHint	This property is used to specify which payment instruments should be available to the buyer. Merchant can specify which features should be supported by payment instrument i.e. later reversal, capture, partial reversal, partial capture etc. Additionally a merchant can request the final step to perform either authorization and capture or authorization alone e.g. CPT:N – only authorization (Manual Capture) CPT:Y – authorization and capture (Auto Capture) (Default behavior) Merchant can also use this property to select one of scenarios which has been configured on Payment Gateway – SCN:<scenario letter> e.g. SCN:X – 'X' is the Scenario ID as configured and communicated by IPG team for a particular type of transaction scenario For controlling whenever and for which brands Payment Portal should require payer to enter Verification Code (i.e. CVV2, CVC2, CID). VCC tag will control verification codes for all brands, while CVV, CVC, CID will control it for specific brand only e.g. VCC:Y – ask verification code for all brands (Default behavior) VCC:N – don't ask verification code for any brand CVV:Y – ask verification code for <i>Visa</i> CVV:N – don't ask verification code for <i>Visa</i> CVC:Y – ask verification code for <i>MasterCard</i> CVC:N – don't ask verification code for <i>MasterCard</i> CID:Y – ask verification code for <i>Discover/AMEX</i> CID:N – don't ask verification code for <i>Discover/AMEX</i> Card Tokenization, whether to apply or not can be defined in the configuration CTK:! – Use CustomerID as seed CTK:X – 'X' will represent a custom seed (Char) for card tokenization CTK:E – Tokenization seed for Etisalat

	Multiple hints within TransactionHint have to be separated by semicolon e.g. pay.SetProperty("TransactionHint", "CPT:N;VCC:Y;SCN:A"); Hints specified by merchant as part of transaction take precedence over those set in merchant profile on payment gateway.
OrderID	This can serve two different purposes; you can either specify an order ID here or ask system to generate one. It can consist of text and special sequences, the total length once the sequences are expanded cannot exceed 16 characters. Special sequences: <u>o Date/Time sequences</u> _ {Y} – year (yyyy format) _ {y} – year (yy format) _ {m} – month (mm format) _ {b} – month (3 letters long abbreviation) e.g. Jun, Feb etc. _ {d} – day (dd format) _ {a} – day of the week (3 letters long abbreviation) e.g. Sun, Mon _ {H} – hour (24 hour system) _ {I} – hour (12 hour system) _ {p} – AM/PM indicator _ {m} – minutes _ {S} – seconds _ {L} – number of milliseconds _ {j} – Julian day (the day number starting from 1st of January) _ {U} – week of the year (assuming Sunday is the first day) _ {W} – week of the year (assuming Monday is the first day) _ {w} – day of week (Sunday=0, Monday=1 etc.)
OrderName	Short description of the order. The OrderName has to be shorter than 25 characters. It can contain only printable Unicode characters and it cannot neither start nor end nor have to adjacent white characters.
OrderInfo	Long description of the goods which are being purchased (this will be displayed on Common Payment Page as a tooltip). The OrderInfo has to be shorter than 256 characters. It can contain only printable and end of line Unicode characters and it cannot neither start nor end nor have to adjacent white characters.

TransactionID	Transaction reference number. This is returned by IPG itself in response of SPI Registration call
---------------	---

5.3.5. Buyer Properties

Property	Description
CardNumber	Credit Card number
ExpiryYear, ExpiryMonth	ExpiryMonth as <i>mm</i> and ExpiryYear as <i>yyyy</i>
VerifyCode	This property refers to CVV2/CVC/etc. The format of the value has to match format used by given card brand (3 digits for Visa/MC/JCB and Diners and 4 digits for AMEX and Simulator). Some brands accept to additional symbols: 'N/A' to indicate that VerifyCode is unavailable and 'ILG' which specifies that the value printed on the card is illegible.
CardTrack2	This property is used in case of card present scenario where payer swaps the card into a machine like KIOSK. KIOSK reads the Card Track 2 Data and sends it to IPG in CardTrack2 field. <i>Note: In case CardTrack2 is being sent to IPG then there is no need to send any other property from Buyer Related Properties.</i>

5.3.6. Response Properties

These response properties can be retrieved in response to a particular call. For code sample/syntax refer to Section 5.3.8.2 Late Bound Properties.

Property	Description
ResponseCode	This field returns the exact response code. Success is always indicated with 0 , and any code using SPI component should verify this status. <i>Note: The exact meaning of this value may be different depend on the buyer's card issuer, merchant's account bank or the step at which authentication failed, which means that we are unable to provide a list of all possible descriptions, in order to receive user-friendly description of the event please use ResponseDescription field.</i> Please refer to FAQ Document for Troubleshooting guide on IPG errors.
ResponseDescription	User-friendly description of the error in ResponseCode. <i>Note: This field can only be used to display the error description and should not be used to check if transaction was successful, to check if transaction was successful please use ResponseCode field.</i>

ResponseClass	This field serves a similar purpose as ResponseCode, but instead of giving a very detailed error it specifies at which stage or part of the system request failed (for example it may point out that issuer declined request or acquire did not accept it or Payment Gateway rejected it, without going into detail)
ResponseClassDescription	This is a user-friendly description of ResponseClass
TransactionID	Unique ID for in progress transaction (Returned in response for every call)
PaymentPage	Link to Common Payment Page where payer will be asked to input Credit Card information for secure transaction (Returned only in response for Registration call)
ApprovalCode	This is the response coming from respective bank for Authorization (Returned only in response for Authorization & Finalization calls)
OrderID	OrderID as provided by Customer or if Customer chose automated OrderID generation in Registration or Authorization call then the generated value (Returned only in response for Authorization & Finalization calls)
Balance	This is the response coming from respective bank for Authorization (Returned only in response for Capture & Reversal calls)

5.3.7. Customer Properties

In addition to the standard set of properties which is available in the *SPI Registration* request it is possible to specify a set of customer-defined properties. **NOTE:** This works for SPI Registration call only.

- Customer properties use the following syntax ExtraData/NameOfProperty
- The first sign of the property name be letter or underscore, while the subsequent ones have to be letter, digit, underscore, hyphen or dot. The total length of 'NameOfProperty' cannot exceed 48 characters.
- The value of property has to follow the same rules as OrderInfo.
- It is customer's responsibility to make sure that each name is unique; if property names are not unique the result is undefined.
- Customer properties can be included in *Registration*, *Authorization*, *Reversal* and *Capture* requests.
- Example:

```
pay.SetProperty ("ExtraData/ShippingAddress", "Mr. John Sheppard P.B. 3838 Abu Dhabi");
```

```
pay.SetProperty ("ExtraData/ContactEmail ", "J.Sheppard@yahoo.comi");
```

5.3.8. Property types

There are two different types of properties and each group is accessed with different syntax.

5.3.8.1. Early bind properties

- Customer
- Store
- Terminal
- Language
- MerchantKeyStore [SPIj only]
- MerchantKeystorePassword [SPIj only]
- TrustedKeystore [SPIj only]
- TrustedKeystorePassword [SPIj only]
- ConnectionAddress
- ConnectionPort
- ConnectionTimeout
- ConnectionSecure
- ResponseCode
- ResponseDescription
- ResponseClass
- ResponseClassDescription

All early bind properties can be accessed using the following syntax.

• SPI net

A = pay.NameOfProperty;

pay.NameOfProperty = A;

For example:

pay.Customer = "ABC";

Resp = pay.ResponseCode;

Connection Related Properties can be accessed using Connection property.

For example:

Address = pay.Connection.Address;

pay.Connection.Address = Address;

• SPIj

A = Object.getNameOfProperty;

Object.setNameOfProperty(A);

For example:

object.setCustomer("ABC");

resp = object.getResponseCode

All properties are strings except:

ConnectionPort, ConnectionTimeout, Language, ResponseCode and ResponseClass which are of type integer and ConnectionSecure which is a boolean value.

5.3.8.2.Late Bind Properties

All other properties like TransactionID, Amount and Currency etc.

All late bind properties can be accessed using the following syntax:

· SPI net

```
A = pay.GetProperty("NameOfProperty")
```

```
pay.SetProperty("NameOfProperty", A)
```

For example:

```
pay.SetProperty("Amount", "1.23")
```

```
TxnID = pay.GetProperty("TransactionID")
```

· SPIj

```
A = Object.getProperty("NameOfProperty")
```

```
Object.setProperty("NameOfProperty", A)
```

For example:

```
object.setProperty("Amount", "1.23");
```

```
TxnID=object.getProperty("TransactionID");
```

6. Configurations and Test Data for Testing in IPG Staging

To assist the merchants for doing POC and sending test transactions to integrate IPG with their system following test configurations and test data shall be used.

6.1. Test Configurations

Following are staging configuration details for Demo Merchant.

Server	demo-ipg.comtrust.ae
Customer ID	Demo Merchant
Channel	Web
Store ID	n/a (this property will is not applicable)
Terminal ID	n/a (this property will is not applicable)
Currency	AED, USD, QAR or PKR
Certificate	Demo Merchant (included in SPI tool kit, can also be downloaded from: https://demo-ipg.comtrust.ae/merchant/downloads/DemoMerchant-Certificates.zip)

6.2. Test Data

Following are staging configuration details for Demo Merchant.

Test Cards	a. 9990000000000003 b. 9990000000000011 c. 9990000000000029
Expiry Month	Any valid month
Expiry Year	Any valid year between 2000 and 2100
Verification Code	Any 4 digit number
Card Brand	IPG

7. FAQs AND KNOWLEDGEBASE

In response to any call posted to IPG through SPI or any other medium like Common Payment Page or SPIless, if request is rejected by IPG or fails to complete at any stage, IPG returns its specific errors. To troubleshoot any errors by IPG please refer to [FAQ Document](#).

8. MERCHANT PORTAL

IPG Merchant Portal sometimes also referred as Web UI, is specially designed to help the customers to view their IPG account online in a Microsoft Silverlight application. The application provides a querying interface to customers to see Registered, Authorized, captured, reversed and declined orders. They can request payment for fulfilled orders, perform adjustments such as reversals and generate reports for transactions.

IPG Merchant Portal can be accessed online at following links:

Production: <https://ipg.comtrust.ae/merchant>

Staging: <https://demo-ipg.comtrust.ae/merchant>

Manual for Merchant Portal can be downloaded from “Downloads” section in Merchant Portal.

NOTE: Merchant Portal is recommended to be accessed using **Internet Explorer** or **Google Chrome** browsers. **Mozilla Fire Fox** is not supported to access transaction details.

9. MERCHANT INTEGRATION TEST CASES

Merchant Integration Support team is keen to provide best support for new IPG merchants in new integrations to IPG. Support team helps the new merchants at every step to make the integration smooth and error free.

Once a merchant is done with successful integration and POC (Proof of Concept) of online payments through IPG using test merchant account “Demo Merchant”, Merchant Integration Support team sets up merchant account on IPG Staging server with exact settings as required by the merchant to be used on IPG Production server; before merchant goes live. Added benefit for this setup is to make sure that there are no issues in IPG integration process and once merchant decides to go live he just has to point to IPG Production server for live transactions.

This section of the document covers the test cases to be executed before going live, to be sure that integration has been done properly hence minimizing any chances of issues and unwanted delays while going live.

Use Case 1: Merchant Configuration 'Registration' confirmation on IPG Staging Server

Use Case	Merchant Configuration 'Registration' confirmation on IPG Staging Server													
Description	Confirmation that the SPI has been configured properly and is able to send correct 'Registration' call Sources: • https://demo-ipg.comtrust.ae/merchant/downloads/IPGManual.pdf • https://demo-ipg.comtrust.ae/merchant/downloads/IPGKnowledgeBase.pdf • https://demo-ipg.comtrust.ae/merchant/ (Using Digital Certificate with Merchant Portal permissions provisioned) • Any user manual for specific SPI version provided with in the SPI download package													
Assumptions	• Developer has downloaded specific SPI version, has gone through the user manuals and has successfully done the integration with Demo Merchant account • Merchant has received Business User Certificate from Etisalat PKI • Merchant has got the Business User Certificate provisioned for transactions													
Actors	• Merchant • Merchant Integration Support													
Steps	• Merchant will send SPI 'Registration' call to IPG Staging server • Merchant Integration Support will verify the received values of the properties sent • Open https://demo-ipg.comtrust.ae/merchant using Merchant certificate, search for the TransactionID returned as a result of successful Registration call; open details of transaction and verify that successful Registration call is in place													
Issues	• Verify/validate the values (properties) sent to SPI at the location where the value is being set into SPI • Verify that ReturnPath is internet accessible link (localhost or local server link cannot be accessed/resolved by client machine when client will be on internet and not on Merchant's local network) • Verify all sent properties to be same as mentioned in the following list <table><tr><th>Property sent to SPI</th><th>Verification Source</th></tr><tr><td>Customer</td><td>Customer ID (Work Order)</td></tr><tr><td>Currency</td><td>Merchant Currency (Work Order) (Please verify currency code supplied from SPI user guide)</td></tr><tr><td>Channel</td><td>Not needed in case only 1 (default) channel has been provisioned</td></tr><tr><td>Store</td><td>Not needed in case only 1 (default) store has been provisioned</td></tr><tr><td>Terminal</td><td>Not needed in case only 1 (default) terminal has been provisioned</td></tr></table>		Property sent to SPI	Verification Source	Customer	Customer ID (Work Order)	Currency	Merchant Currency (Work Order) (Please verify currency code supplied from SPI user guide)	Channel	Not needed in case only 1 (default) channel has been provisioned	Store	Not needed in case only 1 (default) store has been provisioned	Terminal	Not needed in case only 1 (default) terminal has been provisioned
Property sent to SPI	Verification Source													
Customer	Customer ID (Work Order)													
Currency	Merchant Currency (Work Order) (Please verify currency code supplied from SPI user guide)													
Channel	Not needed in case only 1 (default) channel has been provisioned													
Store	Not needed in case only 1 (default) store has been provisioned													
Terminal	Not needed in case only 1 (default) terminal has been provisioned													

	Certificate Path	Is this pointing to same as received from Etisalat PKI?
	Language	Check language codes from SPI user guide
	IPG Connection Address	demo-ipg.comtrust.ae
Comments	Incase anything is confusing or still getting any issues while making Registration call, please contact merchant-integration@etisalat.ae (Merchant Integration Support)	

Use Case 2: Common Payment Page (CPP) appears with correct settings and values

Use Case	Common Payment Page (CPP) appears with correct settings and values
Description	Common Payment Page appearing as a result of 'Registration' contains/loads the requested and provisioned payment instrument, card brands and other related settings Sources: https://demo-ipg.comtrust.ae/merchant/ (Using Digital Certificate with Merchant Portal permissions provisioned)
Assumptions	Merchant has been able to make a successful Registration call and has verified Use Case 1
Actors	Merchant
Steps	- Once CPP loads and is ready for data entry Merchant will verify that he sees all the payment options as of Work Order, for example, Card Brands etc. - Enter valid data and click 'Pay' button to ensure nothing goes wrong or unexpected - Open https://demo-ipg.comtrust.ae/merchant using Merchant certificate, search for the TransactionID returned as a result of successful Registration call; open details of transaction and verify that successful PaymentPage Query call is in place
Issues	- Failure to load CPP, IPG introduction page appears. This normally happens in case a valid TransactionID has not been provided at the time of redirection to CPP - All or some card brands not shown as of Work Order - Detect any unwanted/wrongly implemented policies/scenarios - Transaction Data not valid as of Registration call - Unable to proceed to next step (Finalization that is initiated after clicking 'Pay button')
Comments	Incase anything is confusing still getting any issues while making payment, please contact merchant-integration@etisalat.ae (Merchant Integration Support)

Use Case 3: Merchant Configuration 'Finalization' on IPG Staging Server

Use Case	Merchant Configuration 'Finalization' on IPG Staging Server
-----------------	---

Description	Confirmation that the SPI has been configured perfectly and is able to send correct 'Finalization' call Sources: • https://demo-ipg.comtrust.ae/merchant/downloads/IPGManual.pdf • https://demo-ipg.comtrust.ae/merchant/downloads/IPGKnowledgeBase.pdf • https://demo-ipg.comtrust.ae/merchant/ (Using Digital Certificate with Merchant Portal permissions provisioned) • Plus any user manual for specific SPI version provided with in the SPI download package
Assumptions	• Use Case 2 has passed successfully
Actors	• Merchant
Steps	• IPG will automatically redirect to the ReturnPath (provided by Merchant in 'Registration' call) • Confirm that TransactionID returned in response is same as received in Registration call • Open https://demo-ipg.comtrust.ae/merchant using Merchant certificate, search for the TransactionID returned; open details of transaction and verify that successful Payment Data Update call is in place (amount field will be blank in all calls till this phase) • Finalization called
Issues	• Verify that ReturnPath is internet accessible link (localhost or local server link cannot be accessed by IPG when IPG would try to redirect for Finalization)
Comments	Incase anything is confusing still getting any issues while making Registration call, please contact merchant-integration@etisalat.ae (Merchant Integration Support)

Use Case 4: Capture/Reversal on IPG Staging Server

Use Case	Capture/Reversal on IPG Staging Server
Description	Confirmation that the transaction has been completed successfully Auto Capture: Amount captured automatically (SKIP THIS USE CASE) Manual Capture: Amount has been blocked only (Transaction is not closed and is waiting to be captured or reversed) and Capture/Reversal have been implemented/configured properly. Sources: • https://demo-ipg.comtrust.ae/merchant/downloads/IPGManual.pdf • https://demo-ipg.comtrust.ae/merchant/downloads/IPGKnowledgeBase.pdf • https://demo-ipg.comtrust.ae/merchant/ (Using Digital Certificate with Merchant Portal permissions provisioned) • Plus any user manual for specific SPI version provided with in the SPI download package
Assumptions	• Transaction had been registered for Manual Capture in Registration call
Actors	• Merchant

Steps	• Open https://demo-ipg.comtrust.ae/merchant using Merchant certificate, search for the TransactionID returned; open details of transaction and verify that successful CC Authorization call is in place along with the amount • Merchant will develop a separate page to close the in progress transactions • Merchant has to keep track of open transactions himself and trigger Capture/Reversal manually or in code (as a separate call). • Other way is to pen https://demo-ipg.comtrust.ae/merchant using Merchant certificate, search for the TransactionID; open the details of transaction, at the bottom of page fields/buttons are available for specific action
Issues	• Transaction registered for Auto Capture instead of Manual Capture, hence not available for the process and is Closed • Amount greater than available amount to be captured has been entered • Currency provided is a mismatch to the one provided in Registration call
Comments	Incase anything is confusing still getting any issues while making Registration call, please contact merchant-integration@etisalat.ae (Merchant Integration Support)

Use Case 5: Verify closing of a transaction on IPG Staging Server

Use Case	Verify closing of a transaction on IPG Staging Server
Description	Confirmation that the transaction has been closed and amount has been captured/reversed Sources: • https://demo-ipg.comtrust.ae/merchant/downloads/IPGKnowledgeBase.pdf • https://demo-ipg.comtrust.ae/merchant/ (Using Digital Certificate with Merchant Portal permissions provisioned)
Assumptions	• All use cases run resulted success
Actors	• Merchant
Steps	• Open https://demo-ipg.comtrust.ae/merchant using Merchant certificate, search for the TransactionID; open details • In case of Auto Capture, make sure there exists only 1 CC Capture call after CC Authorization call with full amount captured at once • In case of Manual Capture, Multiple CC Capture and CC Reversal calls can be in place in accordance to the number of Capture/Reversal calls sent to IPG • Make sure that transaction status (second text field in first row of fields at top of page) is 'closed'
Issues	• Transaction faced error at any stage in transaction life cycle and got failed
Comments	Incase anything is confusing still getting any issues while making Registration call, please contact merchant-integration@etisalat.ae (Merchant Integration Support)